OLIVETTI ELEA 9003

MANUALE BASE DI PROGRAMMAZIONE

INDICE GENERALE

INDICE DELLE TAVOLE

INDICE DELLE FOTOGRAFIE

PREMESSA

Il presente manuale ha lo scopo di descrivere le caratteristiche basilari dell'elaboratore Elea 9003, e di fornire gli elementi necessari alla compilazione dei programmi relativi.

Oggetto quindi dei primi capitoli sono la struttura logica funzionale del sistema e la dettagliata descrizione degli elementi che lo compongono.

Nei restanti capitoli vengono analizzati gli elementi che interessano la programmazione: le istruzioni, le possibilita' offerte dai due canali di flusso, dai supporti magnetici collegati, e dalla utilizzazione in linea di apparecchiature particolari di introduzione ed estrazione dei dati.

Non ci si sofferma invece nel presente manuale sulla descrizione tecnica e funzionale delle unita' in linea e fuori linea che restano oggetto di una trattazione specifica.

CAP. 1° : DESCRIZIONE GENERALE DELL'ELABORATORE

1.1. Caratteristiche generali

L'elaboratore elettronico aritmetico Elea 9003 e' uno strumento automatico di grandi capacita' per il trattamento delle informazioni aziendali e per la risoluzione di problemi matematici, scientifici e tecnici; esso consente un ciclo di lavoro interamente automatico, essendo l'intervento umano strettamente limitato alla alimentazione delle informazioni e alla raccolta degli elaborati.

Esso e' costituito da un insieme di apparecchiature che consentono di preparare per la loro successiva elaborazione una grande quantita' di dati, di eseguire quindi su di essi in maniera automatica elaborazioni matematiche, logiche o contabili di qualsiasi tipo, di fornire con rapidita' i risultati nella forma richiesta per l'utilizzazione immediata e l'archiviazione.

Non e' solamente un calcolatore elettronico, nella accezione comune del termine, ma comprende anche le unita' periferiche, i collegamenti delle unita' fra loro e con l'elaboratore centrale, e le apparecchiature fisicamente lontane dal centro di calcolo, ma profondamente integrate con esso, che forniscono le informazioni su cui operare.

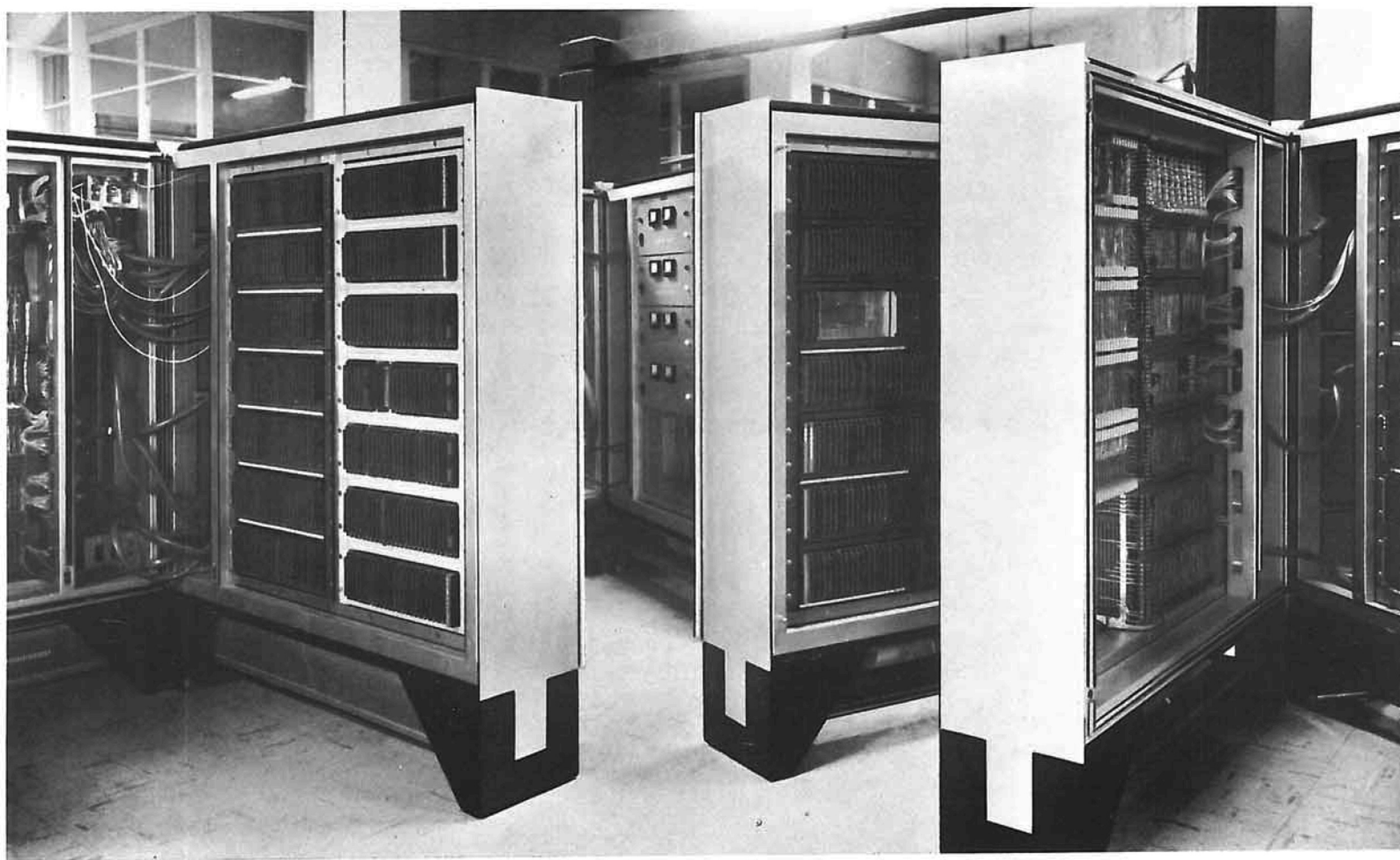
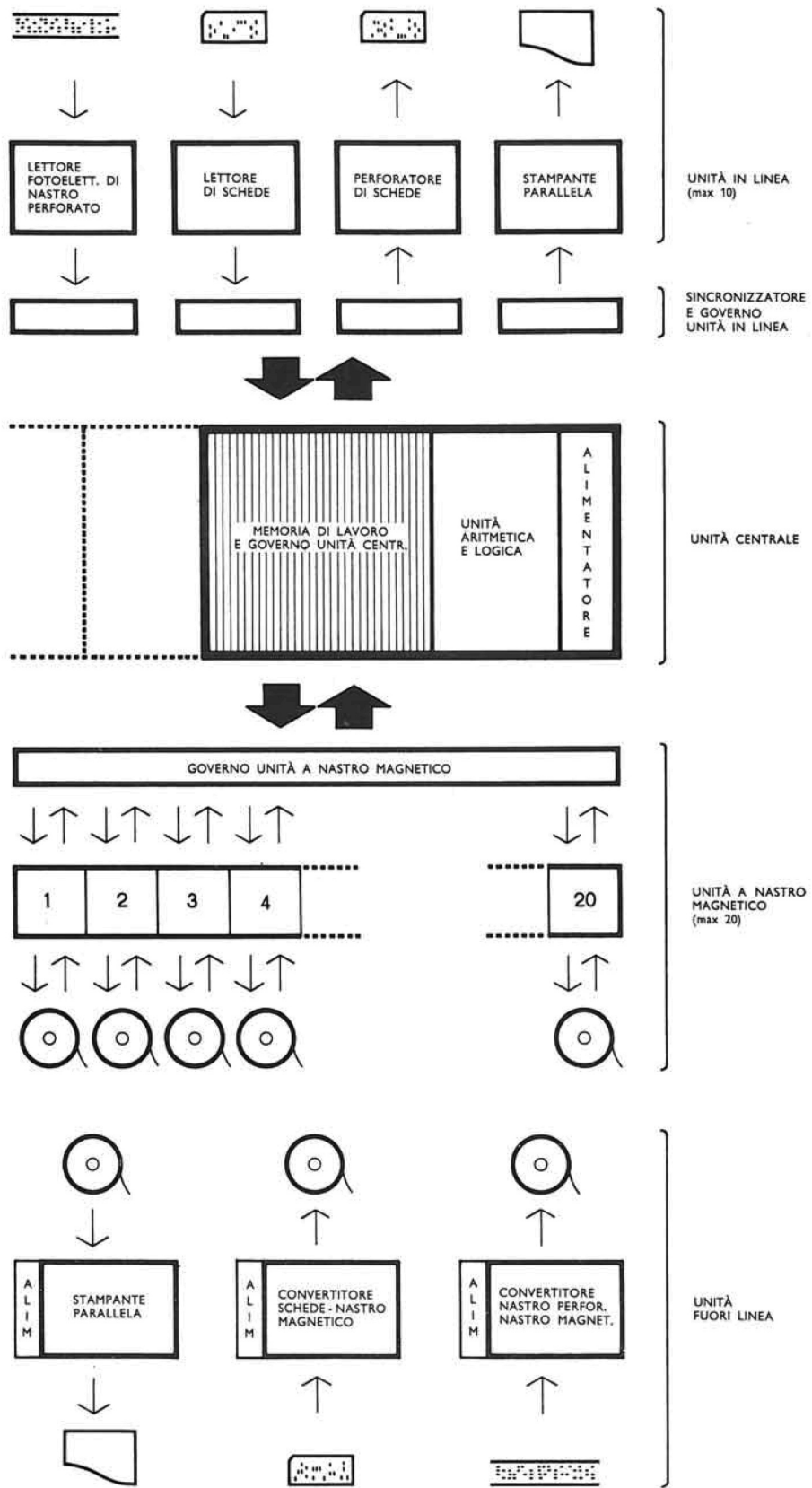
La parte centrale del sistema e' costituita da due unita' : il calcolatore numerico universale, od unita' centrale propriamente detta, e il governo delle unita' a nastro magnetico.

Il calcolatore numerico e' addetto alla elaborazione; e' pertanto questa unita' che, seguendo le istruzioni preparate "una tantum" dall'uomo, ela-

bora tutte le informazioni e controlla contemporaneamente tutte le altre unita' del complesso. In essa, ad esempio, un fatto contabile o amministrativo, che e' stato rilevato all'atto stesso della sua impostazione e successivamente interpretato, e' elaborato e correlato ad altri fatti in maniera del tutto automatica, affinche' la situazione dei vari settori di un'azienda sia costantemente aggiornata, e siano inoltre disponibili elementi sicuri per le decisioni future.

Il governo delle unita' a nastro magnetico costituisce invece la centrale di memorizzazione dell'intero complesso : il nastro magnetico e' infatti il mezzo normale per l'entrata, l'uscita e l'archiviazione dei dati, ed e' il supporto di informazioni piu' conveniente dal punto di vista della capacita' e della flessibilita'.

Il governo delle unita' a nastro magnetico non solo controlla il funzionamento delle unita' a nastro, ma realizza il coordinamento di queste tra loro e l'unita' centrale di calcolo, ed e' in grado anche di eseguire operazioni su archivi magnetici funzionando in modo autonomo. Esso risolve infatti problemi di ricerca su nastro e di aggiornamento di archivi senza interessare l'unita' centrale.

Le altre unita' che compongono il sistema Elea 9003 sono quelle che forniscono le informazioni registrate sui supporti di volta in volta piu' convenienti (nastro perforato, scheda perforata, nastro magnetico, ecc.) e quelle che danno la visualizzazione dei risultati, mediante la compilazione di prospetti o la perforazione di schede.

ELEMENTI AD ARMADIO DEL SISTEMA ELEA 9003 .-

SCHEMA LOGICO DEL SISTEMA ELEA 9003.

1.2. La struttura modulare

La struttura modulare, che caratterizza l'intero sistema, permette di adeguare la potenza dell'Elea al reale bisogno dell'utente, e cio' da' completa garanzia di servirsi sempre di una macchina attuale : i problemi possono cosi' essere visti non in senso statico, bensi' in senso dinamico. All'inizio, la potenzialita' della macchina e' determinata in funzione dei problemi che e' necessario risolvere; in seguito, aumentando il volume del lavoro, si possono collegare nuove unita', e, se cambia la natura dei problemi, si puo' ricorrere ad unita' di tipo diverso per rendere l'elaboratore piu' adatto alle nuove esigenze. Queste prerogative consentono all'utente di evitare continue riorganizzazioni del centro di elaborazione.

1.3. La codificazione dei caratteri

Un insieme di caratteri numerici, alfabetici o speciali, che ha un significato compiuto e che individua una determinata funzione (un codice, un importo, una descrizione, ecc.), costituisce una informazione.

Per la rappresentazione delle informazioni, nell'interno dell'elaboratore, e' stata scelta la forma alfanumerica decimale codificata in binario.

I caratteri alfabetici e le cifre decimali sono quindi rappresentati per mezzo di configurazioni in binario o "bit", in modo da poter sfruttare, nella costruzione degli elementi circuitali, tutti i vantaggi di semplicita' e sicurezza offerti dalla logica a due valori. Il bit, infatti, non e' altro che una variabile che puo' assumere solo due valori opposti, che potrebbero indicarsi con le parole "no", "si", oppu-

re con le cifre "0", "1"; i metodi impiegati per rap presentare materialmente un bit variano a seconda del tipo di organo interessato.

Il codice binario utilizzato nell'Elea 9003 prevede, per ogni carattere, sei bit, i quali consentono di formare 64 configurazioni diverse e cioe' tutte le lettere dell'alfabeto, le cifre decimali e numerosi altri caratteri tra cui i segni algebrici e di interpunzione.

Per la rappresentazione delle informazioni nei supporti di entrata e di uscita si usa il linguaggio che di volta in volta risulta piu' conveniente in funzione del tipo di supporto e della struttura delle informazioni su cui operare.

1.4. I supporti di introduzione e di estrazione

Le informazioni possono essere introdotte nel calcolatore da nastro magnetico, da schede o da nastro perforato, o per mezzo di una tastiera manuale di interrogazione; esse possono essere estratte su nastro magnetico, su schede o su nastro perforato, o su moduli a stampa ottenuti per mezzo di una telescrivente o di stampanti parallele.

Il nastro magnetico occupa una posizione preminente fra i supporti di informazioni utilizzati. Esso infatti consente elevatissime velocita' di introduzione e di estrazione, e la concentrazione di un grande numero di caratteri in un peso e in un volume ridotti. Ha inoltre il vantaggio di poter essere utilizzato un numero praticamente illimitato di volte.

Oltre ad essere il mezzo piu' idoneo per l'introduzione e l'estrazione delle informazioni e dei risultati, presenta anche grandi vantaggi come supporto

per l'archiviazione. Poche bobine sono infatti sufficienti per conservare decine di milioni di dati.

I vantaggi sono rilevanti non solo nei confronti dei tradizionali ingombranti archivi di documenti cartacei, ma anche nei confronti degli archivi di schede perforate.

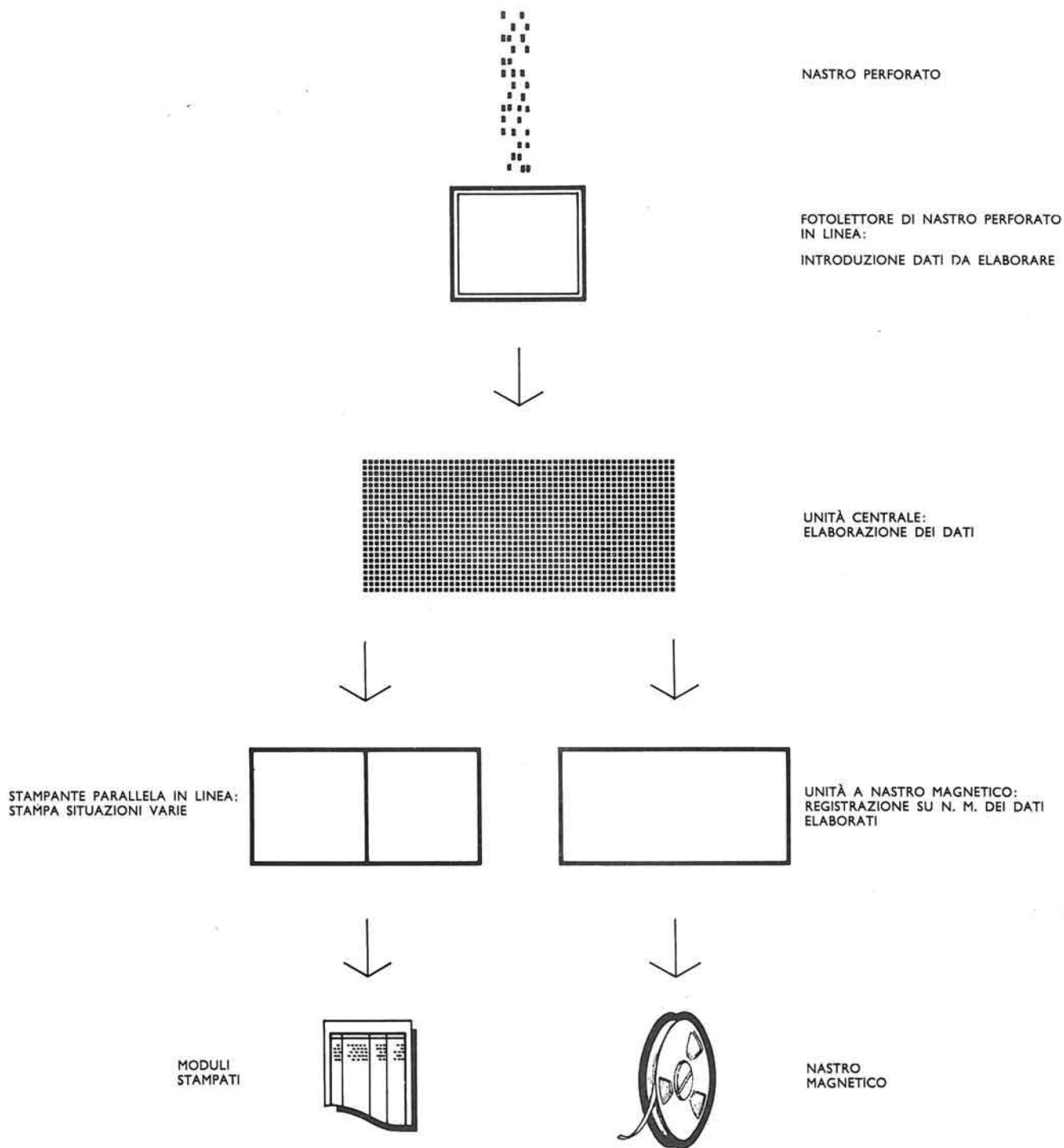
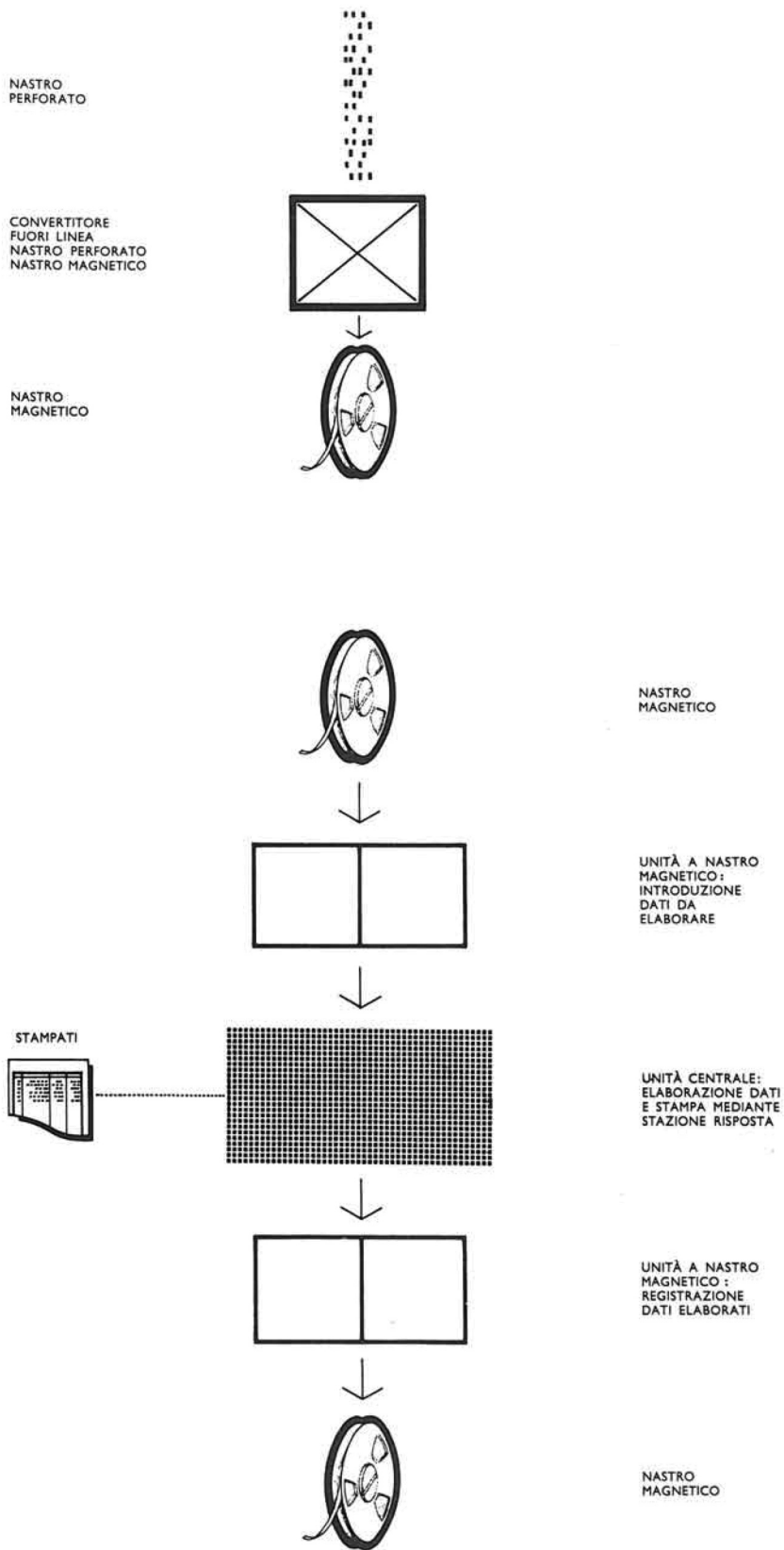
Oltre a cio' la facilita' con la quale si puo' ottenere in pochi minuti la duplicazione di un nastro consente di tenere in doppio tutti i documenti piu' importanti.

Altro supporto di informazioni utilizzato nel sistema Elea 9003, e', come si e' detto, il nastro perforato.

La configurazione geometrica e le caratteristiche strutturali del nastro perforato sono state studiate in modo da rispondere nella maniera migliore alle esigenze di un mezzo particolarmente adatto alla raccolta e alla trasmissione dei dati dalla periferia al centro elettronico.

Il nastro perforato si presenta come una striscia di carta sulla quale i caratteri vengono rappresentati mediante una codificazione che sfrutta 6 canali di perforazione.

Il nastro a 6 canali presenta il vantaggio della larghezza ridotta (la banda e' larga 20,5 mm.) e consente le 64 possibilita' di codificazione previste nel sistema.

Si noti pero' che essendo l'Elea 9003 una macchina "aperta", si possono utilizzare nastri perforati aventi caratteristiche diverse da quelle descritte.

ESEMPIO DI FLUSSO DELLE INFORMAZIONI NEL SISTEMA ELEA 9003: I.

ESEMPIO DI FLUSSO DELLE INFORMAZIONI NEL SISTEMA
ELEA 9003: II.

Anche la scheda perforata, elemento base degli impianti meccanografici tradizionali, puo' essere utilizzata per introdurre ed estrarre i dati direttamente dall'Elea 9003.

Il numero dei supporti collegabili e' variabile conformemente al genere e all'ampiezza dei problemi da risolvere.

Esistono unita' di entrata e di uscita non direttamente collegate all'elaboratore. Si tratta di unita' di conversione delle informazioni da un supporto ad un altro.

In questo caso il collegamento con l'unita' centrale e' assicurato per mezzo di bobine di nastro magnetico su cui vengono registrate le informazioni introdotte dagli altri supporti, o che contengono i risultati da trasferire ad esempio su schede o su nastro perforato o su moduli stampati.

All'elaboratore inoltre possono accedere informazioni provenienti dai piu' disparati organismi di una azienda che siano dotati di macchine contabili e per scrivere munite di apparecchiature per la perforazione. In questo caso durante la compilazione dei documenti si possono ottenere gli stessi dati sotto forma di nastro perforato nel linguaggio accessibile all'elaboratore.

Per la sua particolare struttura l'Elea 9003 puo' infine funzionare oltre che con unita' a nastro magnetico, lettori e perforatori di schede e di nastro perforato, con altre unita' diverse come ad esempio, memoria ad accesso causale, lettori di documenti , ecc.

Queste caratteristiche contribuiscono a far si che l'elaboratore non abbia solo un impiego limitato, ma sia universale, atto a risolvere problemi di qualsiasi tipo.

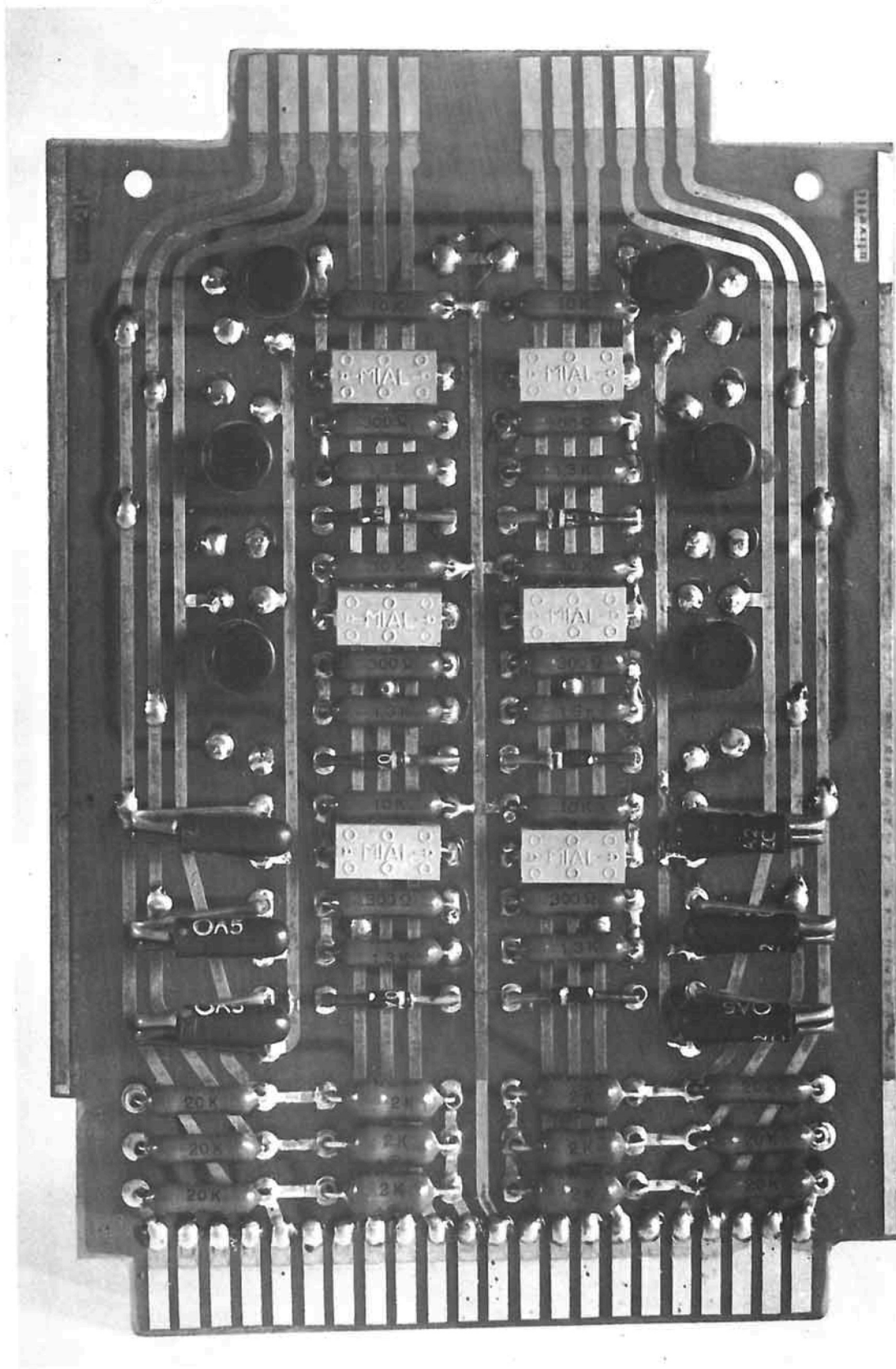
1. 5. Caratteristiche tecnologiche del sistema

La sicurezza operativa dell'Elea 9003 e' favorita dalla moderna tecnologia costruttiva seguita per la sua realizzazione.

I componenti attivi usati sono infatti esclusivamente transistori, gli organi di immagazzinamento delle informazioni sono costituiti da nuclei di ferrite, le funzioni logiche sono realizzate con diodi al germanio.

I circuiti dell'intero complesso sono di tipo standardizzato, stampati su piastrine inseribili, cosi' da rendere facile ed economica sia la costruzione che la manutenzione ordinaria e straordinaria dell'elaboratore.

I collegamenti fra le varie unita' sono ottenuti mediante guide metalliche aeree nelle quali sono contenuti i cavi che assicurano il collegamento fra tutti i circuiti particolari del sistema.

Le caratteristiche tecnologiche su descritte e l'impiego dei circuiti aerei che evitano la necessita' di lavori di adattamento dei locali, permettono i- inoltre di ottenere notevoli risparmi sui costi di esercizio.

PIASTRINA A CIRCUITO STAMPATO.—

CAP. 2° : CARATTERISTICHE LOGICHE DELL'UNITA' CENTRALE

2.1. Caratteristiche generali

L'unita' centrale del sistema Elea 9003, intesa come insieme di organi interni, ha il compito di ricevere le informazioni dalle unita' di introduzione, di elaborarle e di predisporle per le unita' di estrazione: in pratica ha la funzione di dirigere il flusso delle informazioni secondo le richieste dell'utilizzatore.

E' quindi l'unita' centrale che riceve gli ordini dall'esterno, cioe' a dire il programma, che lo fa eseguire impegnando via via tutti gli elementi richiesti.

Gli organi che consentono all'unita' centrale di eseguire questi compiti sono :

- l'unita' di governo
- la memoria di lavoro
- l'accumulatore
- l'unita' aritmetica e logica
- il tavolo di comando

2.2. Il programma

La successione delle operazioni da eseguirsi dall'elaboratore e' determinata dal "programma" registrato nella memoria di lavoro: questo e' formato da una serie di parole di otto caratteri, chiamate istruzioni.

Il programma, e' svolto eseguendo successivamente le istruzioni partendo da quella indicata all'avvio sul tavolo di comando; speciali registri di "indirizzo i-

struzioni" contengono l'indirizzo dell'istruzione in corso e avanzano automaticamente non appena questa sia stata eseguita.

Questo modo di procedere sequenzialmente puo' essere variato mediante apposite istruzioni, dette istruzioni di salto, che permettono di alterare la normale sequenza sia sistematicamente, sia in funzione di eventi rilevati durante l'elaborazione, sia a seguito di condizioni impostate sul tavolo di comando. L'indirizzo dell'ultima istruzione eseguita, prima di passare ad un'altra sequenza puo' essere ricordato automaticamente e senza perdita di tempo, cosi' da facilitare il ritorno alla sequenza iniziale.

Il programma puo' essere registrato in una zona qualsiasi della memoria principale : questa puo' infatti contenere indifferentemente in ogni sua posizione dati da elaborare, risultati e istruzioni di programma.

E' possibile eseguire parallelamente fino a tre sequenze di programma. I tre programmi possono essere iniziati contemporaneamente, ed eseguiti secondo un sistema di priorita' del tutto automatico; tale priorita', naturalmente, entra in giuoco solo quando viene richiesto simultaneamente l'intervento di uno stesso organo da parte di due programmi.

In questo modo possono essere eseguite contemporaneamente :

. una operazione aritmetica o di trasferimento interno,

. una operazione di stampa per mezzo della stazione di risposta alle interrogazioni,

. una ricerca automatica su nastro magnetico,

. una o piu' operazioni di riavvolgimento di nastri magnetici,

- una o piu' operazioni di introduzione od estrazione, per mezzo di unita' a schede o nastro perforato e di stampatrici parallele collegate.

Una tipica utilizzazione dei tre programmi e' la seguente:

- il primo contiene la maggior parte delle istruzioni riguardanti gli organi dell'unita' centrale,
- il secondo contiene le istruzioni riguardanti le unita' a nastro e a tamburo magnetico,
- il terzo le istruzioni relative alle unita' a schede perforate e alle stampatrici parallele direttamente collegate.

E' interessante notare che il terzo programma fa si che queste unita' di lettura - stampa - perforazione impegnino in misura minima il calcolatore.

Qualora esse vengano fatte funzionare insieme ad una unita' a nastro magnetico, consentono la conversione nastro perforato - nastro magnetico, schede perforate - nastro magnetico e nastro magnetico - stampa.

In tal modo puo' dirsi che l'intero complesso e' in grado di funzionare come convertitore senza che lo svolgimento delle elaborazioni ne venga alterato.

2.3. L'unita' di governo

Cuore dell'elaboratore e' l'unita' di governo; essa dirige tutte le operazioni, comanda l'unita' aritmetica e logica, la memoria principale, l'introduzione e l'estrazione dal sistema.

I compiti assolti dall'unita' di governo per eseguire un'operazione sono: il prelevamento in memoria

delle istruzioni da eseguire, la interpretazione di queste istruzioni, l'avvio dei vari organi dell'elaboratore per svolgere le loro funzioni e il controllo di quanto e' stato eseguito. Inoltre il governo e' in grado di tenere conto delle segnalazioni che i vari organi centrali e periferici possono dare per variare il corso delle operazioni.

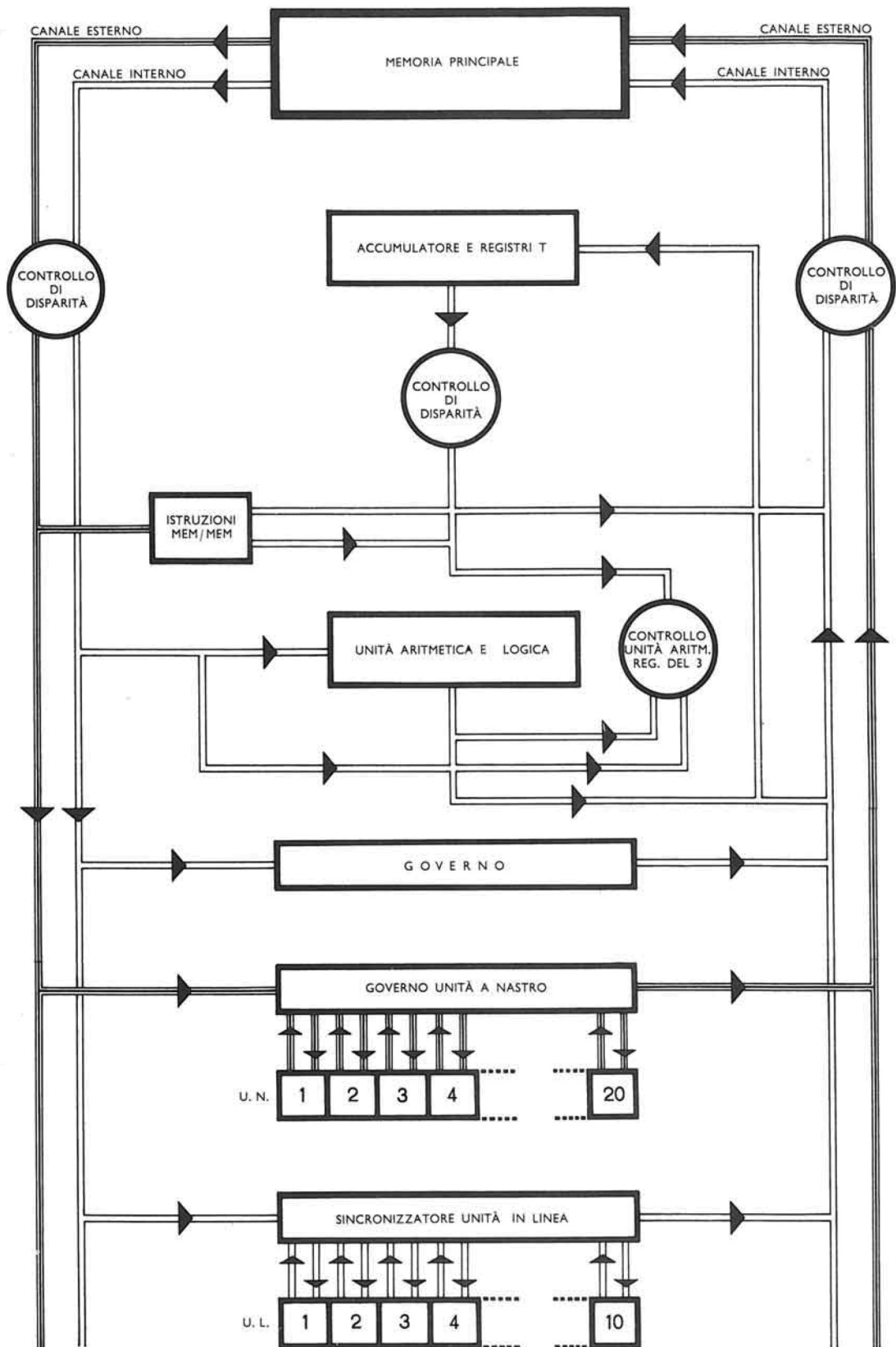
2.4. Il flusso delle informazioni

Il collegamento fra le diverse parti componenti l'unita' centrale e' assicurato da due canali di trasferimento, uno dei quali, detto "canale interno", serve fondamentalmente a collegare la memoria principale all'unita' aritmetica e logica, all'accumulatore e registri T, al governo del calcolatore e sincronizzatore, mentre l'altro, detto "canale esterno" e' utilizzato normalmente per il collegamento della memoria con le unita' di governo dei nastri ed eventualmente dei tamburi magnetici.

Ognuno dei due canali e' provvisto di organi per la verifica di disparita', e di un "indirizzatore" che consente di ricercare nella memoria principale l'indirizzo del carattere che deve essere trasferito o registrato. I due canali possono pertanto lavorare in parallelo, prelevando o registrando nella memoria principale due caratteri alfanumerici per ogni periodo di cifra. Questa possibilita' viene normalmente utilizzata per ottenere la contemporanea esecuzione di operazioni interne di introduzione od estrazione; vi sono pero' istruzioni che sfruttano il parallelismo dei due canali per sovrapporre operazioni di lettura e di registrazione, o per trasferire direttamente una informazione da una zona all'altra della memoria.

La funzione dei due canali e' descritta graficamente nella tabella N. 1 che mette in evidenza anche i collegamenti dell'accumulatore e registri T e degli operatori aritmetico e di verifica.

IL FLUSSO DELLE INFORMAZIONI

Tabella N. 1

L'accumulatore e i registri T sono collegati alla memoria principale dal canale interno, e agli operatori aritmetico e di verifica da un proprio canale, dotato di verifica di disparita'.

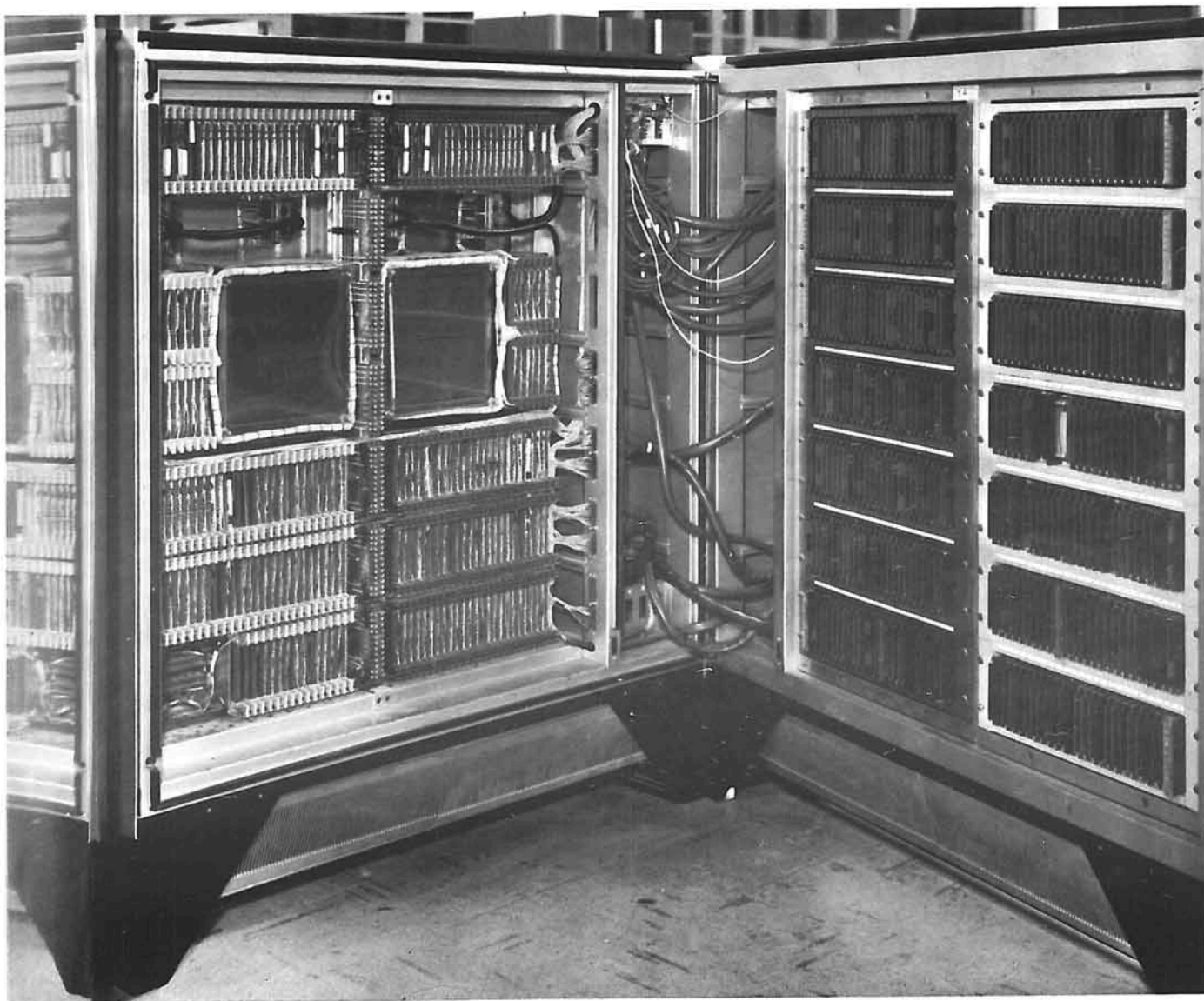
Gli operatori aritmetico e di verifica sono collegati alla memoria principale dal canale interno, attraverso il quale ricevono il secondo operando quando il primo si trova nell'accumulatore o nei registri T. Il calcolatore puo' tuttavia effettuare operazioni su operandi contenuti entrambi in memoria, che arrivano allora agli operatori aritmetico e logico l'uno attraverso il canale esterno, l'altro attraverso il canale interno. Il risultato viene inviato in memoria dall'operatore aritmetico attraverso il canale interno.

2.5. La memoria di lavoro

Tutte le informazioni in entrata vengono convogliate alla memoria di lavoro. Da essa sono prelevate le informazioni in uscita; in essa sono considerati i risultati intermedi, il programma, le costanti di immediata consultazione, ecc.

La memoria di lavoro, o memoria principale e' quindi un punto di passaggio per tutte le informazioni.

Ciascuna posizione della memoria principale e' indirizzabile: cio' significa che il programmatore ha la possibilita' di indicare su quale informazione vuole operare, specificandone semplicemente l'ubicazione . Si possono indicare tanti indirizzi diversi quanti sono i caratteri contenuti.

E' bene definire un'altra suddivisione della memoria, utile da un punto di vista funzionale: i gruppi di caratteri che fanno parte di uno stesso insieme operativo, e che vengono operati per mezzo di un'unica istruzione, sono chiamati "parole". Una parola, per e

LA MEMORIA DI LAVORO IN UN ARMADIO .—

sempio, e' un numero decimale composto da piu' cifre e segno, oppure un nome o una data.

L'indirizzo iniziale e la lunghezza sono gli elementi per identificare una parola; la lunghezza puo' essere espressa mediante un numero di due cifre, oppure identificata, senza bisogno di speciali istruzioni, da un particolare carattere di fine parola. Questo e' scelto dal programmatore e puo' coincidere con il segno algebrico della parola adiacente; in questo caso non vengono occupate posizioni di memoria per delimitare le parole di lunghezza sconosciuta. L'Elea 9003 e' dunque una macchina a parole di lunghezza variabile senza limitazioni.

2.6. L'accumulatore e i registri T

L'accumulatore e' una piccola memoria ausiliare a nuclei magnetici, la cui funzione principale e' quella di contenere uno degli operandi e, successivamente, il risultato di un'operazione aritmetica. La sua capacita' e' di 100 caratteri alfanumerici piu' il segno. In modo analogo alla memoria di lavoro, si puo' indirizzare la parola a partire da una qualsiasi delle posizioni dell'accumulatore. Inoltre, grazie alla presenza di uno speciale "bit" in aggiunta a quelli necessari per formare il carattere, puo' essere segnalata la fine della parola in esso contenuta.

Altra memoria ausiliare a nuclei magnetici e' quella dei registri T: essa ha la capacita' di 200 caratteri alfanumerici indirizzabili di cinque in cinque posizioni per un totale di 40 registri. La loro funzione principale e' la modifica automatica delle istruzioni. Essi pero' possono essere utilizzati per operazioni aritmetiche su operandi la cui lunghezza

non superi 10 caratteri. Esiste uno speciale "bit" analogo a quello dell'accumulatore, che segnala la fine della parola contenuta in un registro.

Gli operandi possono essere trasferiti dalla memoria principale all'accumulatore e viceversa; dalla memoria principale ai registri modificatori e viceversa; ma non direttamente dall'accumulatore ai registri e viceversa; il tempo di trasferimento dipende dalla lunghezza della parola.

2.7. L'unita' aritmetica e logica

L'unita' aritmetica e logica effettua i calcoli aritmetici, i confronti e le operazioni logiche; modifica le istruzioni per mezzo dei registri T; puo' intervenire nei trasferimenti delle informazioni.

Dell'unita' aritmetica e logica fanno parte :

- l'operatore aritmetico e logico,
- l'operatore di verifica,
- il confrontatore.

Le operazioni aritmetiche possono avvenire fra parole dotate o prive di segno; ogni operazione puo' dunque essere algebrica o fra valori assoluti.

Gli operandi elaborati dall'unita' aritmetica e logica non necessariamente devono trovarsi in due memorie diverse (memoria princ.-accumulatore ; memoria principale - registri) ma possono essere contenuti in zone diverse della memoria principale, senza interessare l'accumulatore o altri organi, con conseguente notevole riduzione di tempo e semplificazione dello svolgimento del programma.

I risultati delle operazioni possono ottenersi nella memoria, nell'accumulatore o nei registri di modifica.

Questa unita' ci permette di eseguire operazioni secondo l'algebra di Boole, consentendoci:

1) l'utilizzazione singola dei bit
2) l'elaborazione di proposizioni logiche

Nelle operazioni di confronto l'unita' aritmetica e logica ricorda se le parole confrontate sono uguali o diverse e, in questo caso qual'e' la maggiore; il confronto puo' essere eseguito fra parole numeriche e alfabetiche e fra parole dotate o prive di segno.

Le caratteristiche di quest'operazione rendono particolarmente facile l'ordinamento alfabetico e la selezione di codici o nominativi di identificazione.

2.8. I controlli

L'esatta esecuzione di tutte le elaborazioni che lo Elea 9003 svolge e' garantita da un insieme di controlli automatici che individuando qualsiasi errore non appena questo si verifichi, ne segnalano l'ubicazione nella macchina.

Il controllo dei dati registrati nella memoria principale e nelle altre memorie ausiliarie e' garantito dall'esistenza, per ciascun carattere, di un bit di controllo, la cui funzione e' quella di garantire che ciascun carattere risulti sempre costituito da un numero dispari di bit . Pertanto il controllo delle informazioni nella memoria si ottiene verificando la disparita' del numero di bit in ciascun carattere.

Lo stesso procedimento di controllo e' applicato in

tutti i trasferimenti all'interno dell'elaboratore. Per l'organo di calcolo i controlli sono effettuati mediante la prova del 3. Analogamente alla prova del 9 essa consiste nel confrontare il resto della divisione modulo 3 del risultato dell'operazione, con il resto della divisione modulo 3 del moltiplicando, moltiplicato per il resto della divisione modulo 3 del moltiplicatore e aumentato del resto della divisione modulo 3 dell'addendo.

Esempio :

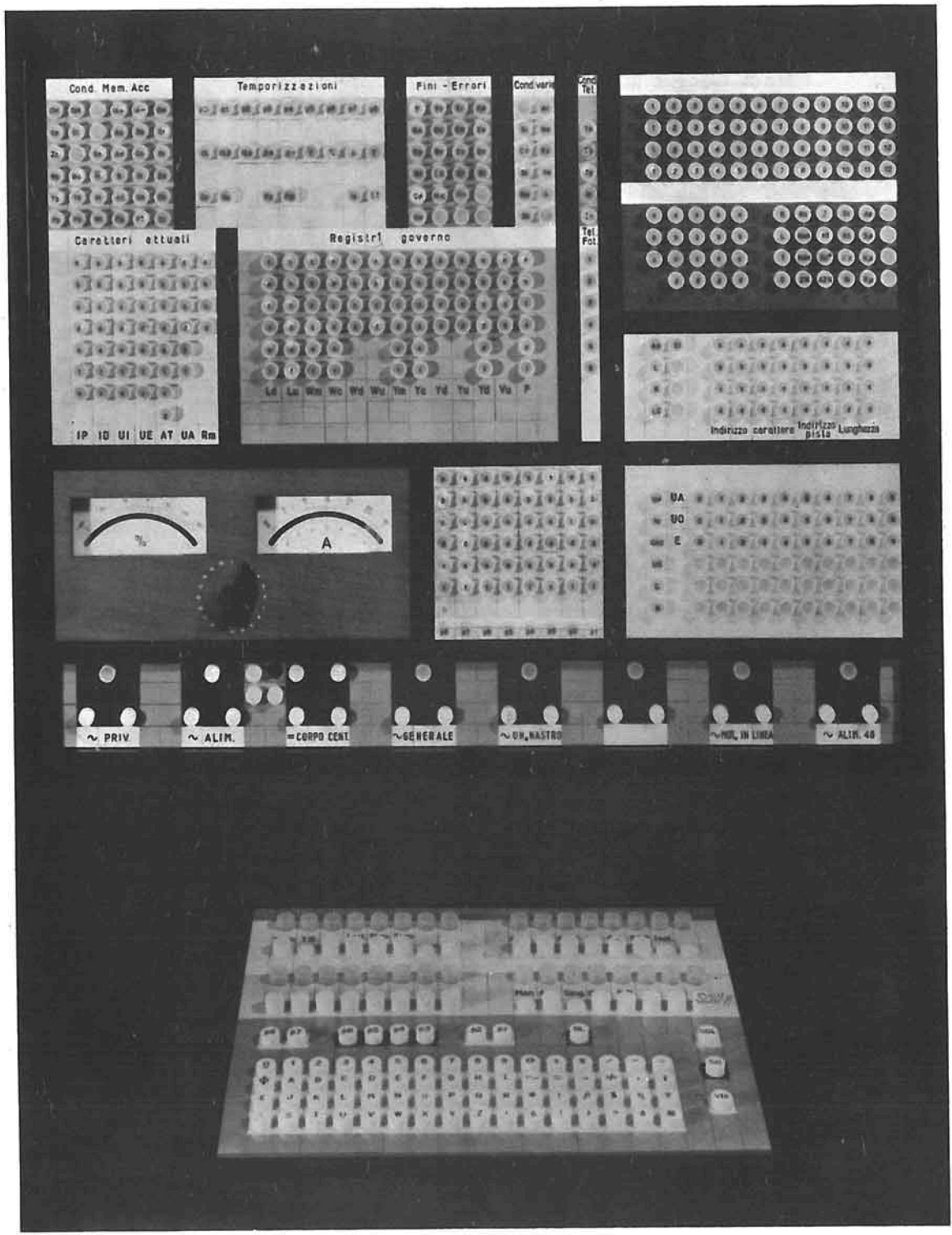
$$19 \times 11 + 6 = 215$$

$$1 \times 2 + 0 = 2$$

Nelle operazioni aritmetiche una ulteriore forma di controllo e' data dalla verifica della rappresentazione dei caratteri numerici ed alfabetici; in essi infatti non compaiono mai le configurazioni c, b, a, = 101, oppure b, a, = 10.

2.9. Il tavolo di comando

Il mezzo di comunicazione fra l'operatore e l'elaboratore e' il tavolo di comando, che contiene i comandi manuali, il quadro e le linee di controllo.

Per mezzo del tavolo di comando si puo' seguire ed eventualmente intervenire nello svolgimento di tutte le operazioni. Esso e' costituito dai tasti e dagli interruttori mediante i quali e' possibile agire sull'elaboratore dall'esterno, e dal quadro di controllo che contiene gruppi di indicatori che permettono di seguire lo stato di avanzamento dell' elaborazione, individuare e localizzare eventuali anomalie.

IL TAVOLO DI COMANDO .—

Al tavolo di comando e' annessa una stazione di risposta alle interrogazioni che ha il compito di prelevare il contenuto delle posizioni desiderate della memoria principale, dandone la trascrizione a stampa; e, qualora lo si desideri, contemporaneamente alla stampa, in modo del tutto automatico e senza perdita di tempo, e' possibile ottenere la perforazione su nastro di carta degli stessi caratteri.

Questa caratteristica risulta particolarmente comoda per la messa a punto dei programmi. La stampa e' completamente indipendente dall'unita' centrale; quest'ultima infatti si limita a fornire i caratteri da stampare e rimane impegnata solamente per il tempo necessario ai trasferimenti che avvengono alla normale velocita' di macchina.

CAP. 3°: CARATTERISTICHE LOGICHE DEL GOVERNO DELLE UNITA' A NASTRO MAGNETICO E DELLE ALTRE UNITA' DI INTRODUZIONE E DI ESTRAZIONE.

3.0. Caratteristiche generali

Il governo dell'unita' a nastro magnetico e' parte costitutiva del sistema Elea 9003. Abbiamo gia' definito il nastro magnetico come il supporto piu' congeniale all'elaboratore elettronico; le unita' relative e il loro governo consentono rispettivamente la sua utilizzazione e il collegamento tra questi e l'unita' centrale.

Alla parte centrale dell'elaboratore e' necessario accedere dall'esterno sia per fare affluire i dati e il programma relativo, sia per estrarre i risultati del conseguente ciclo di elaborazione. Queste due funzioni d'introduzione dei dati e d'estrazione dei risultati, sono affidate alle unita' a nastro magnetico che possiamo quindi definire "unita' d'introduzione" e "unita' di estrazione" del sistema. Esse costituiscono la parte periferica che completa la struttura dell'elaboratore Elea 9003.

La principale caratteristica di queste unita' e' di poter essere collegate alla parte centrale del sistema sia direttamente come unita' in linea, sia indirettamente come unita' fuori linea.

In quest'ultimo caso esse restano staccate fisicamente e logicamente dall'unita' centrale e hanno lo scopo di trasferire i dati da nastro perforato o da scheda perforata, a nastro magnetico eseguendo l'opportuna conversione di linguaggio.

Funzione di questi convertitori e' anche quella di eseguire l'operazione inversa da quella ora descritta

di portare cioe' i dati da nastro magnetico a scheda o nastro perforato, e a stampa.

3.1. Governo unita' a nastro magnetico (GUN)

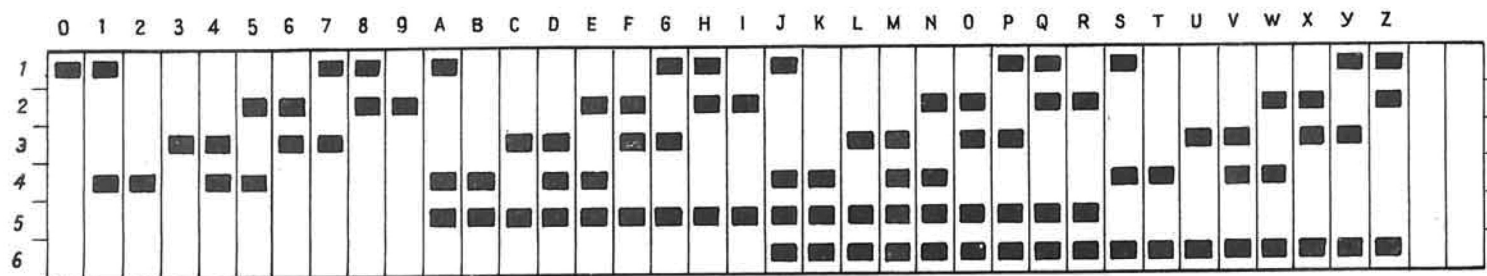
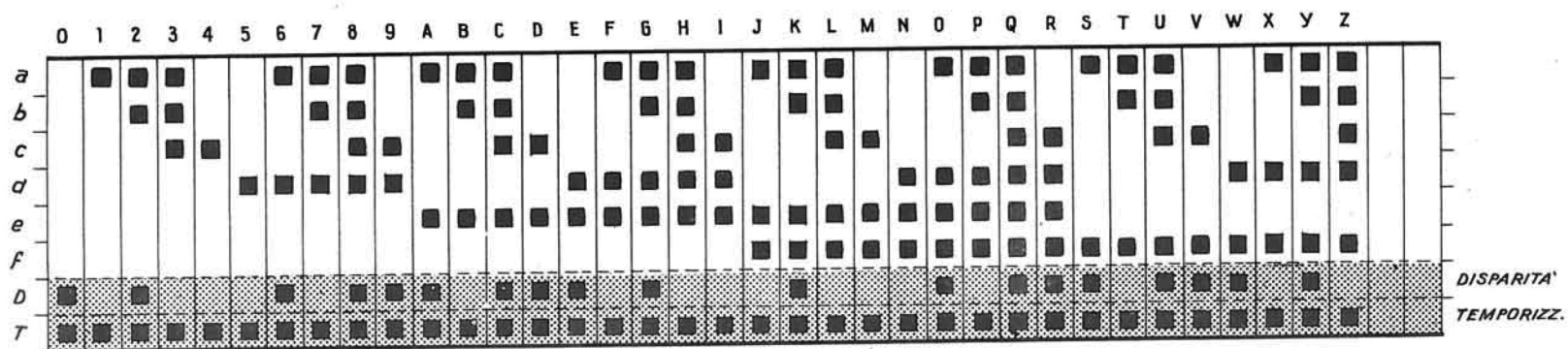
Fino a 20 unita' a nastro magnetico possono essere collegate ad un apposito governo (che costituisce il tramite fra esse e il calcolatore), le cui funzioni sono molteplici ed importanti. Il governo delle unita' a nastro, interamente transistorizzato, assicura e controlla il trasferimento delle informazioni dai nastri magnetici alla memoria principale, e viceversa. Questo trasferimento puo' avvenire consecutivamente oppure selettivamente, e interessare una o piu' unita' a nastro.

Le operazioni di lettura e di registrazione su nastro magnetico oltre che simultaneamente, possono essere eseguite infatti in modo selettivo, nel senso che e' possibile registrare di seguito su nastro il contenuto di zone di memoria non contigue e contemporaneamente suddividere le informazioni in entrata nelle zone di memoria piu' adatte all'elaborazione.

Il governo delle unita' a nastro puo' anche ordinare e controllare la trascrizione delle informazioni da una unita' a nastro, in lettura, ad un'altra unita' in registrazione; e cio' per un numero prefissato di informazioni, o fino a quando non viene trascritta una informazione determinata.

E' possibile inoltre la ricerca di una particolare informazione registrata su nastro magnetico.

Detta operazione puo' avvenire contemporaneamente alla trascrizione del nastro su un altro nastro,

LE CONFIGURAZIONI DEI CARATTERI SU NASTRO MAGNETICO E SU NASTRO PERFORATO

con evidente vantaggio, quindi, nelle operazioni di aggiornamento di un archivio, specialmente quando sia basso il rapporto tra variazione e contenuto dell'archivio stesso. Oltre a cio', il governo delle unita' a nastro puo' ordinare il riavvolgimento contemporaneo di una bobina in un senso o nell'altro per una determinata lunghezza, ecc.

La lettura viene controllata per mezzo della verifica di disparita'; la registrazione invece viene verificata rileggendo attraverso la testina di lettura cio' che e' stato appena registrato attraverso la testina di registrazione, ed eseguendo quindi il confronto.

Nel caso di disuguaglianza, le registrazione continua, ma l'elaboratore denuncia l'errore, che viene corretto subito dopo automaticamente.

3.2. Le unita' in linea

Il collegamento di queste unita' con l'unita' centrale e' stato ottenuto mediante appositi organi le cui funzioni sono di interpretare gli ordini trasmessi dal governo centrale, di coordinare e sincronizzare le operazioni che le singole unita' devono eseguire.

Ogni unita' (stampante parallela, lettore o perforatore di schede, lettore di nastro perforato) e' quindi dotata di un proprio governo, mentre il sincronizzatore e' unico per tutte le unita' e fa parte del sistema centrale.

Sincronizzatore

Il sincronizzatore e' connesso da un lato all'unita' centrale e dall'altro a tante unita' in linea quan-

te sono previste nella particolare installazione per un massimo di 10. Esso contiene anche tutta la logica comune al gruppo di apparecchiature connesse e necessarie allo smistamento di informazioni fra le varie unita'.

Esso dispone di 10 bocchettoni di entrata ed uscita numerati da 1 a 10.

Ad uno qualsiasi di questi bocchettoni puo' essere connessa una qualsiasi unita' meccanica.

Dal momento in cui essa e' connessa ad un certo bocchettone, il numero di ordine di questo diviene l'indirizzo dell'unita' suddetta e con questo indirizzo il programmatore da' qualsiasi indicazione che la riguardi.

Data l'assenza di ogni differenziazione per i bocchettoni, e' possibile collegare ad essi qualunque tipo di unita' entro il limite di 10: come ad esempio 10 stampanti oppure 10 perforatrici, oppure 10 lettori di schede oppure 10 lettori di banda.

Naturalmente nell'uso normale sono collegate simultaneamente unita' di tipo diverso.

Lettore di schede

Il lettore di schede puo' leggere schede a 80 colonne, con qualsiasi tipo di codice di perforazione, alla velocita' di 500 schede al minuto; e' dotato di due caselle di alimentazione e quattro caselle di ricezione.

La scheda si presenta sotto forma di cartoncino indeformabile, reso elettricamente isolante con opportuno trattamento, ed avente forma rettangolare.

Essa puo' essere perforata in diverse posizioni, disposte in 12 linee a 80 colonne. Su ognuna di queste 80 colonne si puo' rappresentare un dato numerico mediante una perforazione posta in una determinata posizione, a seconda della cifra che si vuol rappresentare ; e' pure possibile la rappresentazione di un carattere alfabetico mediante due perforazioni nella stessa colonna, secondo particolari codificazioni.

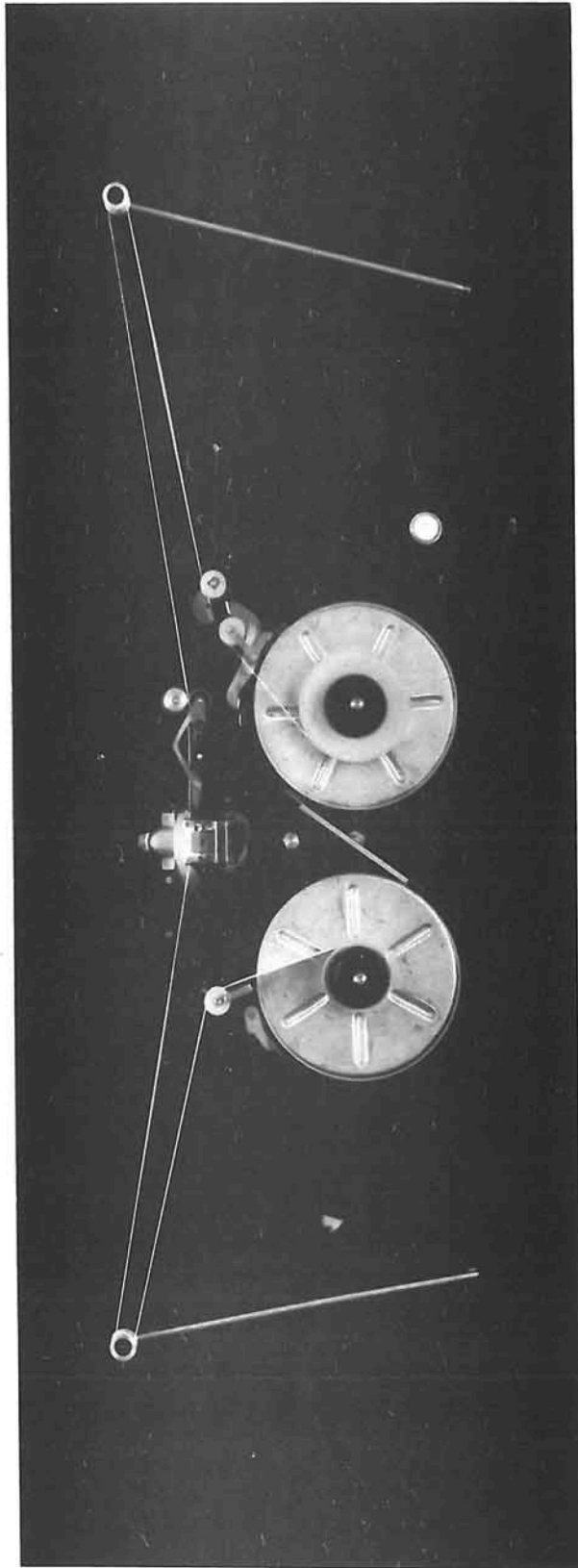
La lettura delle schede e' eseguita da una apparecchiatura fornita di due spazzole a 80 posizioni; la prima spazzola e' utilizzata per la lettura vera e propria. la seconda per la verifica dell'esatta lettura. Con una sola istruzione di programma si specifica l'unita' interessata, si ordina la lettura di una scheda, la registrazione delle informazioni nella memoria di transito e il trasferimento delle informazioni stesse dalla memoria di transito alla memoria principale, nelle posizioni specificate dall'istruzione. Tutte queste operazioni sono rigorosamente controllate.

Perforatore di schede

Il perforatore di schede puo' perforare schede ad 80 colonne, alla velocita' di 150 schede al minuto: e' dotato di una casella di alimentazione, di due caselle di ricezione e di una stazione di lettura a valle della stazione di perforazione.

L'esatta perforazione e' verificata rileggendo le schede, dopo che la perforazione e' stata eseguita, per mezzo di una stazione di lettura a 80 posizioni.

Stampante in linea

E' collegata all'unita' centrale e serve a stampare ad alta velocita' le informazioni elaborate provenienti dalla memoria principale. Le modalita' di stampa possono essere determinate sia dall'unita' centrale, sia

IL FOTOLETTORE .—

dal governo proprio della stampante .

L'operazione di stampa non ostacola il processo di e laborazione.

Il blocco di stampa si compone di 102 ruote di stampa capace ciascuna di stampare 36 caratteri alfanume rici.

La velocita' e' di 300 righe al minuto. La memoria di transito contiene 104 caratteri alfanumerici preleva ti dalla memoria principale.

Prima di essere utilizzate per la stampa, tutte le informazioni vengono controllate e, qualora si rilevi un errore, questo viene posto in evidenza mediante la stampa di un carattere convenzionale ai margini del foglio.

Tutte le unita' in linea possono essere dotate di pannelli di connessione.

Fotolettore

Questo organo permette di registrare direttamente nel la memoria principale del calcolatore le informazioni contenute su banda perforata, alla velocita' di 800 caratteri al secondo. Ad esso si ricorre normalmente per introdurre programmi da mettere a punto, ma puo' essere utilizzato vantaggiosamente per l'introduzione di piccole quantita' di informazioni.

La forma rettangolare dei fori di perforazione (1,65x 2,05 mm) consente il raggiungimento delle piu' alte velocita' agli apparecchi di fotolettura.

Infatti a parita' di dimensione assiale (il lato per

la sezione rettangolare, il diametro per la sezione tonda) tra fori di forma diversa, il fascio luminoso gode nel nostro caso di una maggiore sezione di indagine fotoelettrica.

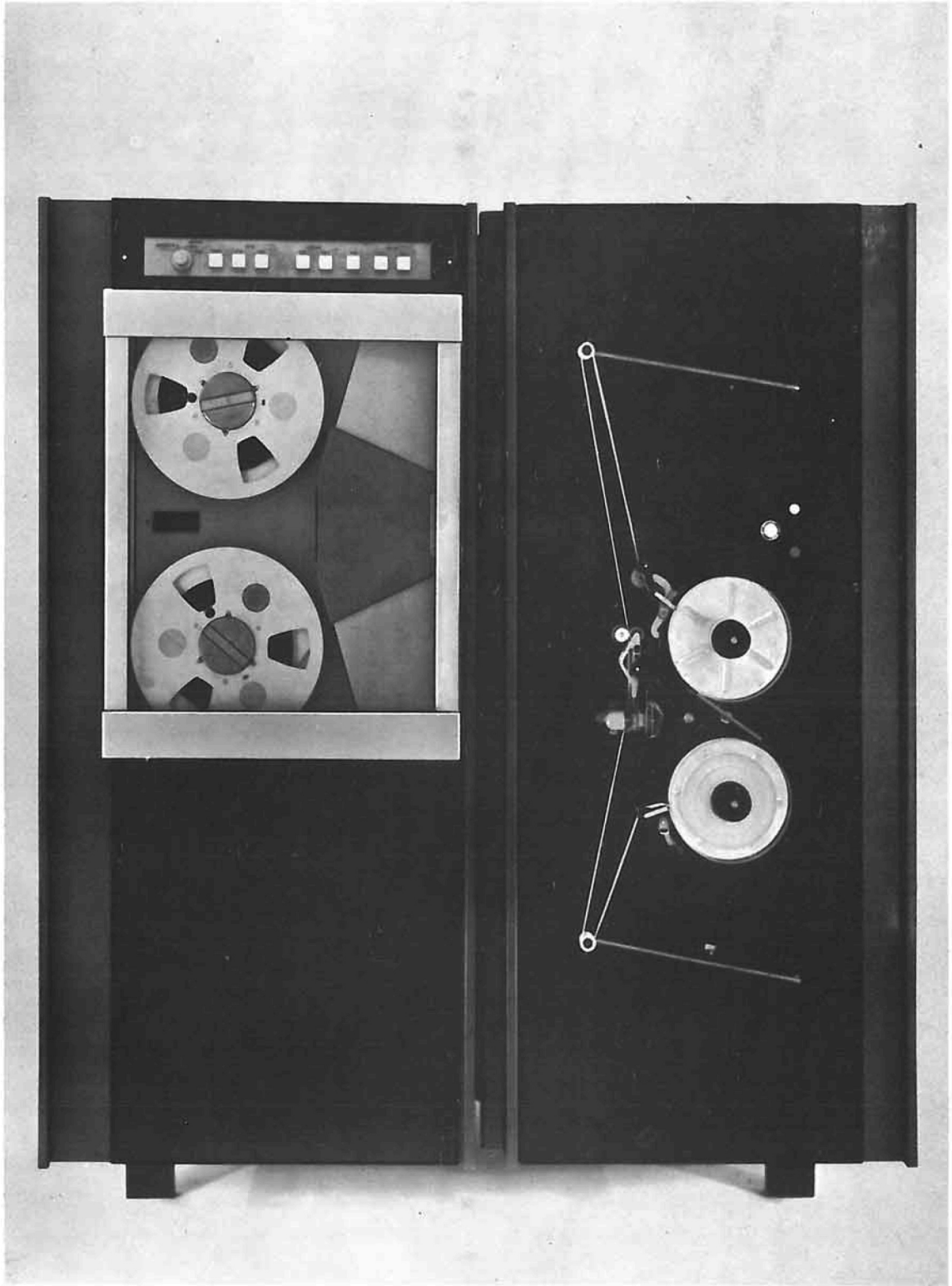
Il convertitore da banda o schede perforate a nastro magnetico

Questa unita' non e' collegata al calcolatore; essa trascrive su nastro magnetico le informazioni contenute su schede o banda perforata, allo scopo di introdurle nel calcolatore per mezzo delle unita' a nastro magnetico, a velocita' maggiore di quella consentita dai lettori di schede e dal fotolettore in linea.

Consta di due fotolettori, capaci ciascuno di leggere 800 caratteri al secondo, e di un lettore di schede che legge 700 schede al minuto; il lettore di schede e' munito di piu' caselle di ricezione, per agevolare l'asportazione delle schede gia' lette.

Le informazioni sono riportate su nastro magnetico per mezzo di una unita' a nastro collegata; esse vengono automaticamente riorganizzate, se necessario, e trascritte a blocchi di lunghezza prefissabile fino ad un massimo di 1200 caratteri.

Pure automaticamente vengono registrati sul nastro magnetico, nelle posizioni opportune, i caratteri di servizio necessari alla sua utilizzazione in fase di elaborazione; si possono inserire sino a 10 costanti diverse, e togliere o aggiungere il segno a una parola numerica.

Il controllo della lettura del carattere in uscita da banda o scheda perforata viene eseguito rileggendo lo stesso carattere per mezzo di due testine fo-

IL CONVERTITORE DA BANDA PERFORATA A NASTRO MAGNETICO .—

toelettriche o di due spazzole di lettura, ed eseguendo quindi un confronto.

Il controllo della registrazione su nastro magnetico viene effettuato mediante il bit di disparita'.

La stampante fuori linea

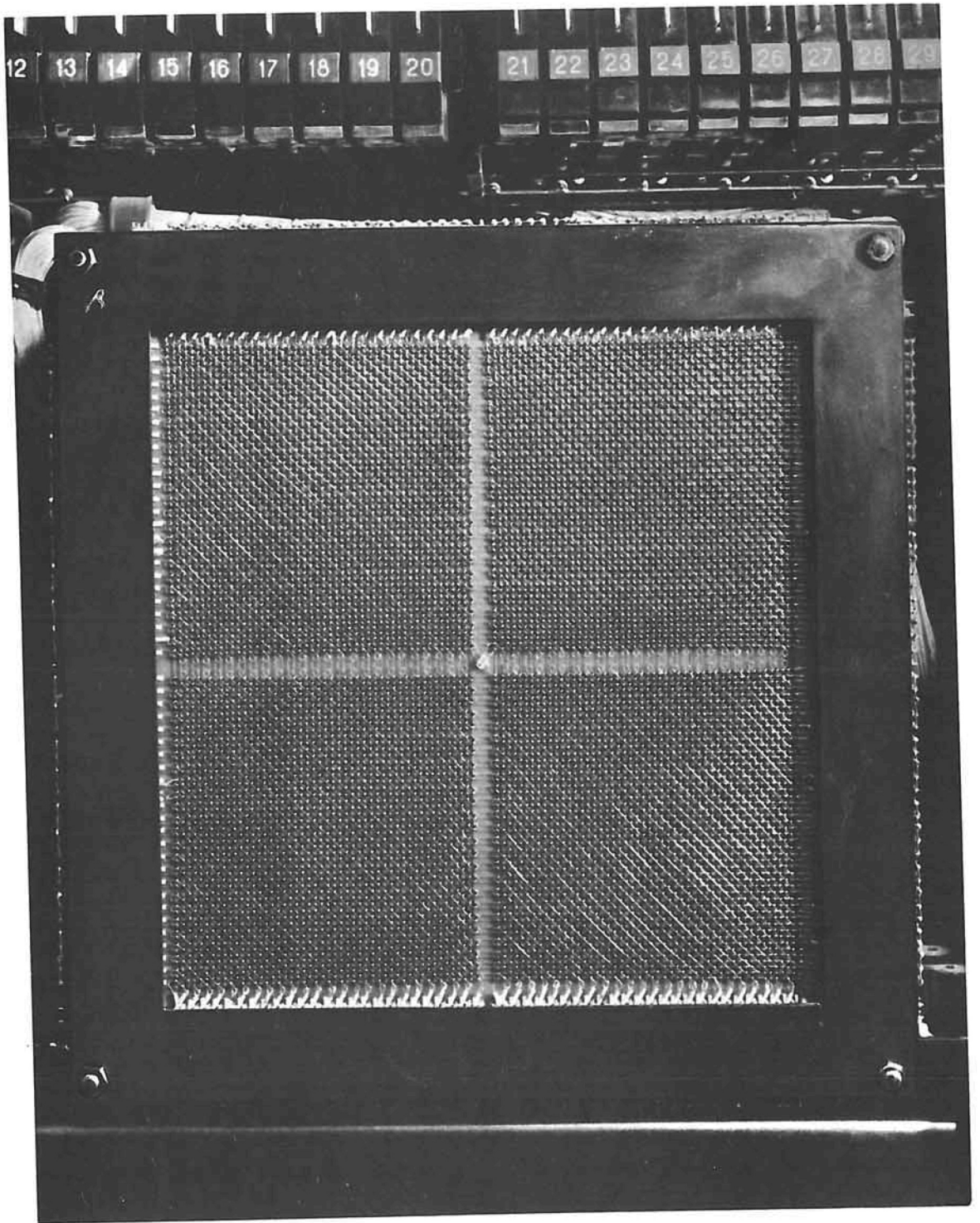
Questa unita' non e' collegata al calcolatore; essa stampa, ad una velocita' variabile da 600 a 1000 righe/minuto, con 120 caratteri/riga, il contenuto dei nastri magnetici ottenuto sia come risultato di elaborazioni effettuate dal calcolatore sia per la conversione da schede o da nastro perforato. Riceve i dati da una unita' a nastro magnetico e provvede alla loro organizzazione per la stampa mediante un pannello mobile di connessione e un lettore fotoelettrico di un anello di nastro perforato.

Le informazioni da stampare sono organizzate su nastro magnetico nel modo consueto e vengono lette e trasferite, un blocco alla volta, nella memoria di transito che ha la capacita' di 2048 caratteri alfanumerici.

Da qui vengono prelevate all'istante voluto e inviate alla stampa tramite un pannello mobile di connessione, che determina sia il tempo sia la destinazione d'uscita.

L'eccezionale flessibilita' del convertitore da nastro magnetico a stampa permette di eliminare, durante l'elaborazione nel calcolatore, il tempo necessario alla organizzazione dei dati per la stampa e di ottenere, dallo stesso nastro, stampati diversi sia orizzontalmente che verticalmente. La lettura del nastro, la registrazione nella memoria di transito, la trans-codificazione, l'avanzamento

della carta sono sistematicamente ed automaticamente controllati sicche' si rende superflua la presenza di un operatore che sorvegli la stampa.

I caratteri stampabili sono: 10 numerici, 26 alfanumerici, e 20 fra i principali simboli matematici, commerciali e di interpunzione. Sono ottenibili fino a 6 copie simultaneamente.

LA MEMORIA DI LAVORO .–

CAP. 4°: L'UNITA' CENTRALE

4.1. Memoria principale

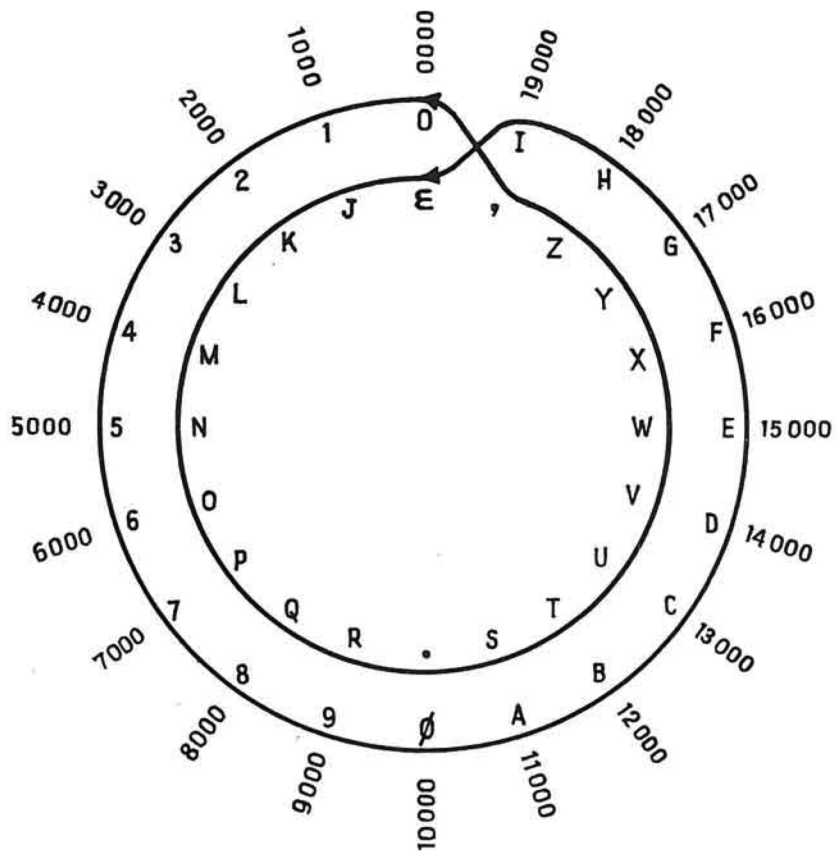
La memoria principale e' costituita da nuclei di ferrite a ciclo di isteresi rettangolare. I nuclei sono montati su piani, ognuno dei quali ne contiene 10.000. Sette piani sovrapposti contengono 70.000 bit necessari a rappresentare 10.000 informazioni alfanumeriche; il loro insieme forma un "elemento" tecnologico di memoria.

Due elementi tecnologici di memoria, per un totale di 20.000 informazioni alfanumeriche, costituiscono invece la minima unita' funzionale di memoria : un calcolatore puo' disporre di un numero di unita' di memoria variabile da 1 a 8, corrispondenti ad un numero di informazioni alfanumeriche compreso fra 20.000 e 160.000.

I sette bit costituenti ogni carattere vengono operati in parallelo: il tempo necessario al loro trasferimento da una zona ad un'altra della memoria, o dalla memoria agli altri organi della unita' centrale e' di 10 microsecondi; tale intervallo di tempo e' detto "periodo di cifra".
In un periodo di cifra si possono estrarre o introdurre dalla memoria due caratteri, ed operare su di essi.

La memoria e' circolare e indirizzabile posizione per posizione mediante indirizzi di quattro cifre la cui aritmetica e' per contare modulo 40.000.

Cio' significa che in una macchina con memoria di 20.000 posizioni qualora si oltrepassasse l'indirizzo 19.999 (I999), in tutti gli organi programmabili (memoria, registri T, comparatori ecc.) apparirebbe-

ro numeri relativi alle posizioni di una memoria avente la capacita' di 40.000 posizioni: cioe' Ɛ000 (20.000), J000 (21.000), K000 (22.000), ecc. Essendo pero' la memoria di 20.000 posizioni e' evidente che per la circolarita' della memoria l'indirizzo Ɛ000 corrisponde alla posizione 0000 e che l' indirizzo J000 corrisponde alla posizione 1000 ecc., come illustrato dalla figura seguente:

Per quanto riguarda le memoria con piu' di 40.000 posizioni vi e' l'impossibilita' di percorrere mediante operazione aritmetica (e modifica automatica) l'intero insieme degli indirizzi.

Accade cioe' che oltrepassando l'indirizzo 39.999 ('999) l'indirizzatore sceglie non l' indirizzo 40.000 (0Ø00) ma l'indirizzo 0000.

Non esiste cioe' la possibilita' di attraversare con operazioni aritmetiche i multipli di 40.000.

Questo non avviene invece per aggiornamenti effettuati nell'ambito di un gruppo di memoria di 40.000 caratteri.

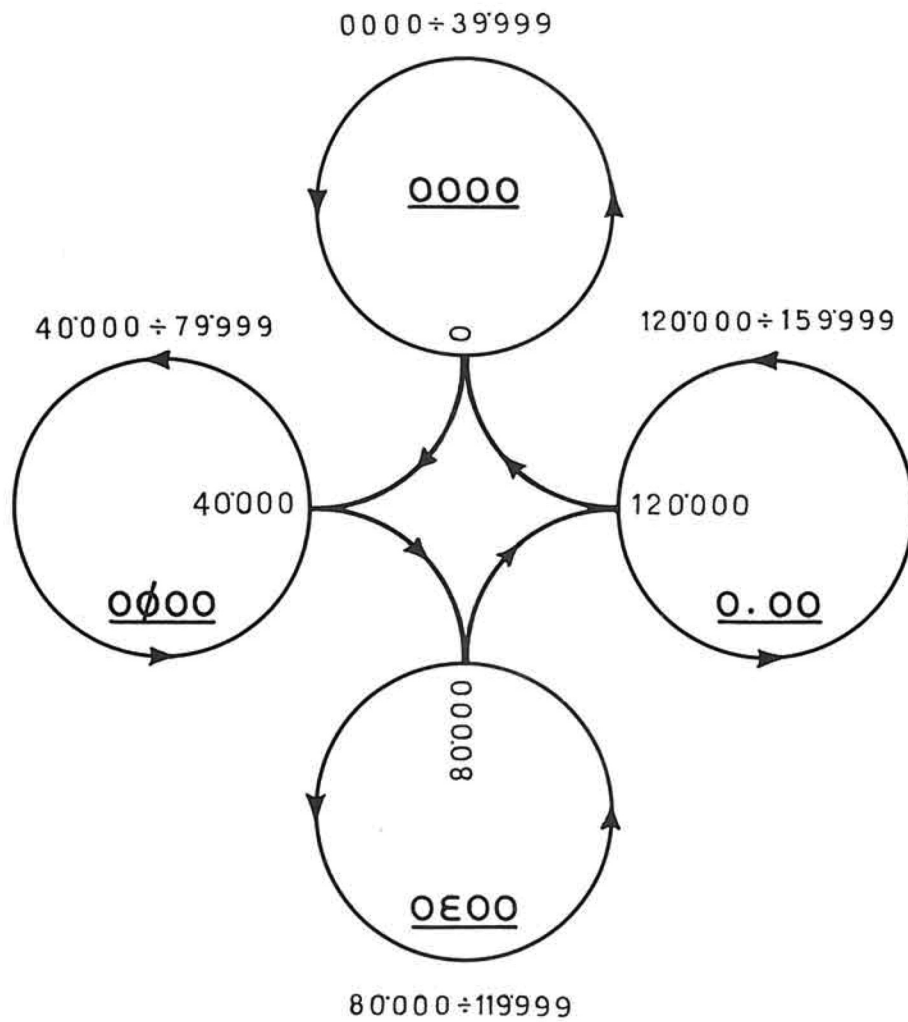
Da cui :

```
( 39.999) '999 + 1 =  0000 (      0)
( 79.999) 'I99 + 1 =  0Ø00 ( 40.000)
(119.999) 'R99 + 1 =  0£00 ( 80.000)
(159.999) ''99 + 1 =  0.00 (120.000)

(      0) 0000 - 1 =  '999 ( 39.999)
( 40.000) 0Ø00 - 1 =  'I99 ( 79.999)
( 80.000) 0£00 - 1 =  'R99 (119.999)
(120.000) 0.00 - 1 =  ''99 (159.999)

( 39.999) '999 - 1 =  '998 ( 39.998)
( 79.999) 'I99 - 1 =  'I98 ( 79.998)
(119.999) 'R99 - 1 =  'R98 (119.998)
(159.999) ''99 - 1 =  ''98 (159.998)

(      0) 0000 + 1 =  0001 (      1)
( 40.000) 0Ø00 + 1 =  0Ø01 ( 40.001)
( 80.000) 0£00 + 1 =  0£01 ( 80.001)
(120.000) 0.00 + 1 =  0.01 (120.001)
```

POSIZIONE O PER OGNI GRUPPO TECNOLOGICO DI MEMORIA

 = CIRCOLARITÀ NELL'AMBITO DI UN ELEMENTO TECNOLOGICO DI MEMORIA.

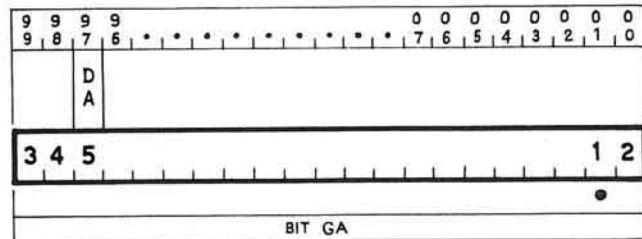
 = ORDINE DI SUCCESSIONE DEGLI ELEMENTI TECNOLOGICI DI MEMORIA.

4.2. Accumulatore

L'accumulatore e' una speciale memoria a 100 posizioni; ad ogni posizione e' associato un indirizzo. Gli indirizzi vanno da 00 a 99; l'indirizzo successivo al 99 e l'indirizzo 00, cioe' l' accumulatore e' circolare. L'indirizzo iniziale di una parola contenuta in accumulatore e' dato dal contenuto di uno speciale registro detto DA.

DA - Mediante l'istruzione DA si puo' fissare l'inizio dell'accumulatore in una qualsiasi delle sue 100 posizioni. Cio' significa che il primo carattere di una parola che venga prelevata o inviata all'accumulatore proverra' o andra' all'indirizzo fissato mediante la istruzione DA. I caratteri successivi hanno riferimento agli indirizzi identificati da numero d'ordine maggiore.

Esempio :

- Accumulatore dopo un trasferimento con DA = 97 del numero di 5 cifre : 12345.
- Si noti l'influenza della circolarita' della memoria.

Bit gA - Il registro DA indica dunque l'indirizzo di una parola contenuta in accumulatore; la fine dell'informazione e' segnata invece da uno speciale bit detto gA. Cio' permette di utilizzare l'accumulatore per contenere piu' parole; ad ognuna di esse corrispondera' un bit gA che ne segnala la fine, mentre il contenuto del registro DA rappresenta l'inizio della parola sulla quale si vuole operare.

I diversi bit gA sono materializzati da un ottavo piano di nuclei ferromagnetici, che si aggiunge ai sette utilizzati per la registrazione dei caratteri alfanumerici e per la verifica di disparita'.

Il bit gA ha i seguenti effetti :

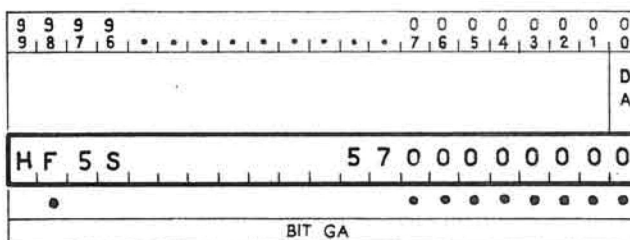
- nelle operazioni aritmetiche le cifre che lo seguono sono considerate degli zeri;

- nelle operazioni di trasferimento da accumulatore a memoria il bit gA non viene considerato.

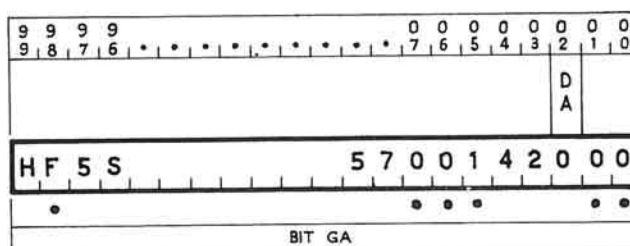
La posizione del bit gA e' determinata dalle seguenti regole generali :

- l'istruzione AoM (trasferimento da accumulatore a memoria con azzeramento delle posizioni di accumulatore trasferite) pone un gA in corrispondenza di ogni posizione che viene azzerata.

Esempio :

- Accumulatore dopo una AoM di 8 posizioni supponendo DA = 00.

- L'istruzione di trasferimento da memoria ad accumulatore (MA) pone un bit gA in corrispondenza dell'ultimo carattere trasferito e lascia inalterati i bit gA posti precedentemente purche' non si riferiscano ai caratteri trasferiti.

Esempio :

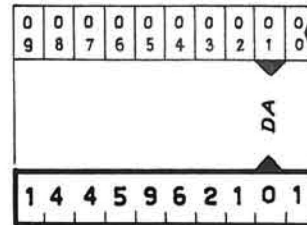
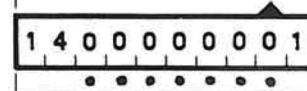
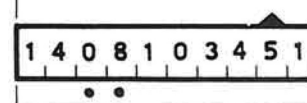
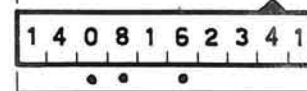
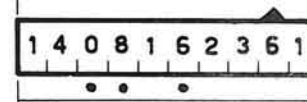
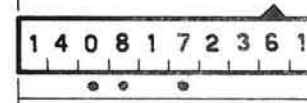
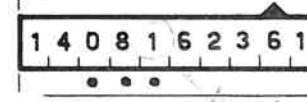
- Accumulatore dopo un AoM (esempio precedente) e un trasferimento con DA = 02, del numero 1420.

- Nelle istruzioni aritmetiche il bit gA viene disposto secondo le seguenti regole :

  . se l'operando chiamato dalla memoria ha una lunghezza minore o uguale al numero di posizioni comprese fra DA e gA, questo non viene spostato;

  . in caso contrario il gA viene posto in corrispondenza della cifra piu' significativa dell'operando chiamato da memoria;

  . nel caso di superamento di capacita' dovuto a riporto, il gA viene posto in corrispondenza della cifra piu' significativa e cioe' eventualmente una posizione piu' a sinistra di quella che risulterebbe diversamente.

L'esempio che segue mostra la posizione del bit gA attraverso una successione di istruzioni :

Condizioni iniziali

dopo una AoM lunga 7

dopo il trasferimento in acc. numero 810345 (lung. 6)

dopo il trasferimento del numero 6234 (lung. 4)

dopo l'addizione del numero 2 (lung. 1)

dopo l'addizione del numero 1000 (lung. 4)

dopo l'addizione del numero 9000 (lung. 4)

Per quanto riguarda le operazioni algebriche il valore assoluto di un numero puo' essere registrato in accumulatore sia nella sua vera grandezza che nel suo complemento alla potenza di 10 immediatamente superiore : un apposito organo a flip-flop segnala con la sua posizione quale delle due rappresentazioni e' in quel momento utilizzata.

La memorizzazione del segno e' invece affidata ad uno speciale registro, chiamato "registro del segno" che segnala se la parola contenuta in accumulatore e' dotata di segno, e in caso affermativo, se il segno e' "piu'" o "meno".

Questo registro e il flip-flop segnalatore dei complementi possono essere considerati come organo unico che puo' indicare una eventualita' fra sei possibili : le due eventualita' "in vera grandezza" e in "complemento" si compongono infatti con le tre eventualita' "non segnato", "segno piu'" e "segno meno".

VERA GRANDEZZA			COMPLEMENTO		
NON SEGNATO	SEGNO +	SEGNO −	NON SEGNATO	SEGNO +	SEGNO −

Altri tre registri concorrono all'esecuzione delle operazioni algebriche (le operazioni aritmetiche possono essere considerate operazioni algebriche tra operandi di segno + : in questo caso pero' i risultati non sono ovviamente segnati).

NORME CHE REGOLANO LO STATO DEL REGISTRO DE SEGNO

VERA GRANDEZZA		COM LEMENTO	
NON SEGNATO	SEGNATO	NON SEGNATO	SEGNATO
a) dopo un trasferimento mediante AOM b) dopo un trasferimento mediante MA di parola non segnata c) dopo una operazione tra operandi non segnati, e risultato in vera grandezza	a) dopo un trasferimento mediante MA di parola segnata b) dopo una operazione tra operandi, di cui uno almeno sia segnato	a) qualora da una operazione tra addendi non segnat si ottenga il risultato in compl mento.	a) qualora da una operazione tra addendi di cui almeno uno sia segnato, si ottenga il risultato in complemento

Essi servono a ricavare rispettivamente :

1) il segno dell'operazione da eseguirsi (somma, sottrazione);

2) il segno dell'operando contenuto in memoria;

3) il segno del moltiplicatore.

Si hanno quindi complessivamente 4 registri e conseguentemente 4 indicazioni di segno per ogni operazione. (Il segno del moltiplicatore non e' considerato nelle operazioni di somma e sottrazione e il segno della moltiplicazione non compare nel registro del segno dell'operazione; in sua vece si hanno i segni + o - ad indicare se il prodotto ottenuto va sommato o sottratto al contenuto dell'accumulatore).

In base al numero complessivo dei segni "-" che possono comparire nei quattro registri, durante una operazione, si hanno le seguenti norme che regolano la complementazione dei dati operandi e dei risultati:

1°) Se il numero dei segni "-" e' "dispari" si ha la complementazione dell'operando contenuto in memoria con inversione di segno prima che su questi si effettui l'operazione richiesta.

2°) Se in seguito alla complementazione del dato in memoria e alla sua inversione di segno si ottiene un risultato con "riporto", il riporto viene annullato e il risultato compare in "vera grandezza". Vedi esempio 1.

3°) Se in seguito alla complementazione del dato di memoria e della sua inversione di segno si ottiene un risultato senza "riporto" il risultato compare in "complemento". Vedi esempio 2.

Esempio 1

NUMERO SEGNI NEGATIVI	OPERANDO MEMORIA	OPERANDO ACCUMUL.	SEGNO OPERAZIONE	MOLTIPLICATORE
1 = Dispari	10 −	20 (N.S)	+	
COMPLEMENTAZIONE	90 +			
SI	RISULTATO =	(1)10 +	VERA GRANDEZZA	

Esempio 2

NUMERO SEGNI NEGATIVI	OPERANDO MEMORIA	OPERANDO ACCUMUL.	SEGNO OPERAZIONE	MOLTIPLICATORE
1 = Dispari	20 (N.S)	10 (N.S)	−	
COMPLEMENTAZIONE	80			
SI	RISULTATO =	90 (N.S)	COMPLEMENTO	

Esempio 3

NUMERO SEGNI NEGATIVI	OPERANDO MEMORIA	OPERANDO ACCUMUL.	SEGNO OPERAZIONE	MOLTIPLICATORE
1 = Dispari	20 (N.S)	20 (N.S)	−	
COMPLEMENTAZIONE	80			
SI	RISULTATO =	(1)00 (N.S)	VERA GRANDEZZA	

Esempio 4

NUMERO SEGNI NEGATIVI	OPERANDO MEMORIA	OPERANDO ACCUMUL.	SEGNO OPERAZIONE	MOLTIPLICATORE
2 = Pari	10 −	20 +	−	
COMPLEMENTAZIONE	10 +			
NO	RISULTATO =	30 +	VERA GRANDEZZA	

4°) Se in seguito alla complementazione del dato di memoria e della sua inversione di segno si ottiene un risultato nullo = 0, il risultato e' indicato in "vera grandezza". Vedi esempio 3.

5°) Se il numero dei segni "-" e' "pari", l'operando di memoria viene operato tale e quale se si tratta di una somma, o col segno inverso se si tratta di una sottrazione. Vedi esempio 4.

Per quanto riguarda il trasferimento di un operando o di un risultato dall'accumulatore a memoria si hanno le seguenti possibilita' :

DATO IN ACCUMULATORE		➡	DATO DOPO IL TRASFERIMENTO	
non segnato	vera grandezza		non segnato	vera grandezza
" "	complemento		" "	complemento
segno +	vera grandezza		segno +	vera grandezza
" +	complemento		" −	" "
" −	vera grandezza		" −	" "
" −	complemento		" +	" "

4.3. Registri T

L'insieme dei registri T e' costituito da una memoria a nuclei magnetici della capacita' di 200 caratteri alfanumerici, indirizzabili di 5 in 5 posizioni. Ad ogni possibile indirizzo corrisponde un registro T; vi sono dunque 40 di tali registri.

Ogni registro ha la lunghezza massima di 10 posizioni; e' chiaro quindi che, ad esempio, il registro 2 ed il registro B si sovrappongono parzialmente.

NOME REGISTRI	I	H	G	F	E	D	C	B	A	Φ	I
INDIRIZZI		90	80	70	60	50	40	30	20	10	0
	95	85	75	65	55	45	35	25	15	05	
NOME REGISTRI	9	8	7	6	5	4	3	2	1	0	

NOME REGISTRI	9	Z	Y	X	W	V	U	T	S	°	9
INDIRIZZI		190	180	170	160	150	140	130	120	110	100
	195	185	175	165	155	145	135	125	115	105	
NOME REGISTRI	R	Q	P	O	N	M	L	K	J	Ɛ	

Quando il registro T e' utilizzato per modificare un'istruzione, o, come si dice, quando opera come modificatore, al massimo i primi 5 caratteri del suo contenuto vengono utilizzati; conseguentemente, tutti i 40 registri possono operare come modificatori senza che risulti alcun inconveniente dalla loro parziale sovrapposizione.

Immaginando per convenienza di esposizione le posizioni di memoria numerate da 000 a 199, la parola registrata a partire dall'indirizzo zero, normalmente uno dei fattori della moltiplicazione, puo' avere una lunghezza qualsiasi fino ad un massimo di 100 caratteri, piu' un segno che viene trasferito in un apposito registro; le parole registrate a partire dagli altri indirizzi possono avere un massimo di 10 caratteri.

Bit gT - Non esiste per i registri T l'analogo della DA : le informazioni contenute in ogni registro hanno sempre inizio dall'indirizzo corrispondente a quel registro. Esiste invece l'analogo del bit gA detto gT; anche per i registri T il bit gT e' materializzato da un ottavo piano di nuclei magnetici .

Esiste ambiparita' anche per quanto riguarda il registro del segno; solo che questo resta accessibile esclusivamente mediante l'istruzione "Y".

Per quanto riguarda l'utilizzazione ed il posizionamento del bit gT valgono le stesse considerazioni fatte a proposito del bit gA ovviamente sostituendo le istruzioni per l'accumulatore con le corrispondenti per i registri T.

Nelle singole istruzioni verra' specificato quando il risultato puo' avere lunghezza piu' lunga del piu' lungo degli operandi ed in quali casi si ha indicazione di overflow.

Un registro T viene identificato nell'istruzione da un carattere; la tabella precedente indica l'indirizzo iniziale di ogni registro T in corrispondenza del carattere che lo identifica.

4.4. Logica aritmetica dell'Elea 9003

Operazioni aritmetiche che possono interessare la memoria, l'accumulatore e i registri T (+ MM ; - MM ; + MT ; - MT ; + MA ; - MA ; + TM ; + AM).

Le operazioni su caratteri speciali non possono essere effettuate. Nel caso si abbiano operazioni su tali caratteri si ha indicazione di errore nell'unita' aritmetica e logica.

Possono invece essere effettuate operazioni su caratteri alfabetici e ovviamente su caratteri numerici.

La macchina sa distinguere questi caratteri dagli altri, in quanto in essi non appaiono le configurazioni c, b, a = 101, oppure b, a = 10 .

Operando su numeri :

a) la parte numerica (*) viene operata normalmente (modulo 10)

b) un eventuale riporto non influisce sulla configurazione della parte alfabetica (*).

Si ha pertanto :

		f e	d c b a	
	4	(0 0	0 1 0 0	)
+	5	(0 0	1 0 0 0	)
=	9	(0 0	1 1 0 0	)

		f e	d c b a
9		(0 0	1 1 0 0)
+ 4		(0 0	0 1 0 0)
= 1 3	(0 0 0 0 0 1)	(0 0	0 1 1 1)

Operando su caratteri alfabetici e alfanumerici :

1° la parte numerica viene operata allo stesso modo dei caratteri numerici, tenendo conto del valore numerico dato alla lettera in questione (vedi tabella 2);

2° i risultati delle operazioni sono espressi mediante uno o due caratteri a seconda che si abbia avuto o no riporto dalla somma dei due caratteri;

3° la parte alfabetica viene operata distintamente dalla parte numerica, secondo modulo aritmetico 4 (da cui 3 + 1 = 0);

4° la complementazione nella parte alfabetica avviene anch'essa secondo modulo aritmetico 4.

Si ha pertanto :

Esempio 1°

		f	e		d c b a	
	D	0	1	(1)	0 1 0 0	(4)
+	D	0	1	(1)	0 1 0 0	(4)
=	Q	1	1	(2)	1 1 1 1	(8)

Esempio 2^{o}

	f	e		d c b a	
D	0	1	(1)	0 1 0 0	(4)
+ I	0	1	(1)	1 1 0 0	(9)
= L	1	1	(2)	0 1 1 1	(3)
1	0	0	(0)	0 0 0 1	(1 di riporto)

	0 0	0 1	1 1	1 0	
0 0	0 0	0 1	1 1	1 0	0
0 1	0 1	1 1	1 0	0 0	1
1 1	1 1	1 0	0 0	0 1	2
1 0	1 0	0 0	0 1	1 1	3
	0	1	2	3	

Tabella delle somme dei bit "f, e".

* Nota : I caratteri numerici hanno i bit "f, e" = 00.

I caratteri alfabetici ripetono nei bit "d, c, b, a" le configurazioni dei caratteri numerici a cui sono associati (vedi tabella 2) e si distinguono pertanto solo per la configurazione dei bit "f, e".

Esempio :

carattere	f e	d c b a
4	0 0	0 1 0 0
D	0 1	0 1 0 0
M	1 1	0 1 0 0
V	1 0	0 1 0 0

identica configurazione

MODULO ARITMETICO					0	1	2	3
BIT	d	c	b	a	f e	f e	f e	f e
					0 0	0 1	1 1	1 0
					CARATTERI			
	0	0	0	0	0	ø	ε	•
	0	0	0	1	1	A	J	S
	0	0	1	1	2	B	K	T
	0	1	1	1	3	C	L	U
	0	1	0	0	4	D	M	V
	1	0	0	0	5	E	N	W
	1	0	0	1	6	F	O	X
	1	0	1	1	7	G	P	Y
	1	1	1	1	8	H	Q	Z
	1	1	0	0	9	I	R	,
	0	0	1	0	~	δ	*	θ
	0	1	0	1	+	=	(	!
CARATTERI NON OPERABILI SECONDO LA LOGICA ARITMETICA	0	1	1	0	#	α	β	)
	1	0	1	0	/	%	$	?
	1	1	0	1	–	,	η	&
	1	1	1	0	÷	γ	π	⊗

Tabella N. 2

CAP. 5°: LE UNITA' A NASTRO MAGNETICO

5.1. Governo delle unita' a nastro

E' questo l'organo mediante il quale si effettua il collegamento tra unita' a nastro e calcolatore, oppure fra due unita' a nastro.

Le unita' collegabili al GUN sono al massimo 20.

Le caratteristiche funzionali del GUN permettono di eseguire la verifica delle registrazioni contemporaneamente alla registrazione stessa e di rilevare e segnalare all'unita' centrale eventuali errori di lettura o di registrazione. Il GUN puo' ordinare il riavvolgimento contemporaneo delle bobine di un numero qualsiasi di unita' a nastro, e durante il riavvolgimento operare su una unita' in lettura ed una in registrazione.

Altre e piu' complesse operazioni possono essere eseguite dal GUN. Esso puo' disporre lo svolgimento di un nastro magnetico, nell'uno o nell'altro senso, fino a far passare un numero di blocchi indicato da un'istruzione di programma; puo' far trascrivere un numero di blocchi indicato da una istruzione, da un nastro in lettura a un nastro in registrazione.

Puo' inoltre ordinare lo svolgimento di un nastro magnetico fino ad arrivare al blocco che contiene una parola di lunghezza variabile previamente trasferita in uno speciale registro. Questa operazione, detta di ricerca, puo' essere effettuata anche trascrivendo contemporaneamente i blocchi che non contengono la parola cercata ad un nastro in registrazione. Essa rende possibile la consultazione ed aggiornamento di un nastro archivio a grande velocita' e senza che sia impegnato il calcolatore.

5.2. Nastro magnetico

Il nastro magnetico e' una banda di materia plastica (mylar) ricoperto su una faccia da ossido di ferro magnetizzabile. Si puo' pensare che la superficie del nastro sia costituita da numerose piccole aree che possono essere magnetizzate o no. Lo stato magnetico di ciascuna di queste aree permette dunque la rappresentazione materiale di un bit.

Degli 8 bit che vengono utilizzati per ciascun carattere, uno serve per il controllo dell' esattezza della registrazione e uno per la temporizzazione.

Ogni carattere numerico, alfabetico o speciale e' quindi rappresentato dalle combinazioni dei 6 bit restanti secondo il codice impiegato dall'Elea 9003.

I caratteri sono registrati uno di seguito all'altro, in gruppi di capacita' variabile, ma normalmente dell'ordine di diverse centinaia o alcune migliaia di caratteri; tali gruppi, detti blocchi, sono separati sul nastro da un breve spazio non utilizzato, detto interblocco. La lunghezza dell'interblocco e' di 1 pollice.

La larghezza del nastro e' di 1,27 cm., la lunghezza di una bobina e' di circa 1100 metri. La densita' di registrazione e' di ~120 caratteri per centimetro (corrispondente a 300 caratteri per pollice). Conseguentemente ogni bobina di nastro puo' contenere fino a 12.960.000 caratteri. quasi 13 milioni !

Il nastro viene svolto nell'uno e nell'altro senso, alla velocita' di 3,81 mt. al secondo; in un secondo vengono letti (o registrati) 45.000 caratteri. Il tempo di lettura di una intera bobina (12.960.000 caratteri) e' quindi inferiore ai 5 minuti.

ALCUNE UNITA' A NASTRO MAGNETICO COLLEGATE FR - 400 .-

Per conoscere il numero N di caratteri registrabili in una bobina di nastro, con blocchi di n caratteri in media, si deve applicare la formula seguente :

$$N = \frac{12.960.000 \cdot n}{n + 300}$$

Cosi' ad esempio se i blocchi contengono 1000 caratteri, il numero totale di caratteri registrati sara' di 9.969.230 .

Il numero b dei blocchi registrabili su una bobina e' dato ovviamente dalla formula analoga :

$$b = \frac{12.960.000}{n + 300}$$

Alle due estremita' della bobina il nastro e' metallizzato per una lunghezza opportuna, per avvertire l'unita' che lo governa della sua prossima fine.

Le unita' a nastro magnetico dell'Elea 9003 sono dotate di due testine, una per la lettura e una per la registrazione.

Esse possono leggere il nastro sia quando questo scorre nel senso detto convenzionalmente "avanti", sia quando scorre nel senso opposto, detto "indietro"; la verifica della corretta lettura e' effettuata per mezzo del bit di disparita'. La registrazione e' invece verificata rileggendo con la testina di lettura, che si trova a valle di quella di registrazione quando il nastro scorre in avanti, quanto appena registrato, e confrontandolo con l'informazione che doveva essere registrata. Come conseguenza, la registrazione puo' avvenire solo nel senso "avanti".

Quando l'unita' riceve l'ordine di iniziare la lettura o la registrazione di un blocco impiega un tempo di 5,7 millisecondi circa per avviare il nastro; terminata la lettura impiega 4,1 millisecondi per arrestare il nastro. L'interblocco di 1 pollice di lunghezza corrisponde appunto alla lunghezza di nastro svolta durante questi tempi di decelerazione ed accelerazione.

Le unita' a nastro sono capaci di riavvolgere la bobina completamente, senza intervento dell'elaboratore, che si limita ad impartire l'ordine relativo.

I caratteri che distinguono le diverse unita' a nastro collegabili al GUN, in numero di 20, sono i seguenti :

1, 2, 3, 4, 5, 6, 7, 8, 9, ~

Ø, A, B, C, D, E, F, G, H, I,

5.3. Organizzazione delle informazioni su nastro magnetico

Il raggruppamento dei caratteri in blocchi, dovuto alla necessita' di trasferire un numero limitato di caratteri dal nastro magnetico alla memoria principale dell'elaboratore, esige d'altra parte - almeno se si vuole che la lunghezza del blocco sia variabile - uno speciale carattere che marchi l'inizio e la fine di ogni blocco. Tale carattere e', per l'Elea 9003, il carattere α : ogni blocco deve iniziare e terminare con α.

Esiste d'altra parte, in molti casi, la convenienza di suddividere in sequenze, comprendenti uno o piu' blocchi, l'intera informazione registrata su nastro: a questo scopo si fara' terminare l'ultimo blocco

della sequenza, ed iniziare il primo, con il carattere "punto interrogativo" (?) invece che con il carattere α.

Per marcare la fine della informazione registrata su nastro magnetico viene invece utilizzato il carattere "moltiplicato per" (X).

Per suddividere un blocco in parti di lunghezza determinata, ai fini di una registrazione o lettura su indirizzi non consecutivi di memoria, viene usato il carattere Θ.

CAP. 6°: ORGANIZZAZIONE DELLA PROGRAMMAZIONE

6.1. Struttura della programmazione

Chiamiamo programma una sequenza di disposizioni o
comandi che eseguiti uno di seguito all'altro, por
tano alla soluzione di un problema secondo la logi
ca particolare dell'elaboratore.

L'impostazione di un programma puo' essere ovvia-
mente diversa secondo le caratteristiche dell'ela-
boratore a cui il programma stesso va indirizzato;
e' indispensabile quindi conoscerne i particolari
requisiti, studiarne a fondo le possibili utilizza
zioni, per riuscire a costruire un programma che
sfrutti al massimo le capacita' della macchina.

Oltre che il numero e la struttura dei comandi che
il sistema Elea 9003 e' in grado di interpretare ed
eseguire, si rende necessario conoscere di esso le
peculiari caratteristiche : la simultaneita' opera
tiva o plurisequenzialita' di programma, e lo sfrut
tamento massimo del nastro magnetico come principa
le supporto delle informazioni.

E' sulla traccia di queste nozioni, che il program
matore deve risolvere i problemi che gli vengono
sottoposti.

La stesura della soluzione prescelta deve pertanto
essere espressa mediante un linguaggio comprensibi
le alla macchina : e' percio' richiesta al program
matore la conoscenza dei simboli con cui i comandi
devono essere formulati e le regole a cui essi so-
no soggetti.

L'esprimersi nel modo sopradetto significa espri-
mersi in linguaggio macchina.

E' tuttavia opportuno notare che esiste il modo di evitare al programmatore la fatica che questo mezzo di comunicazione comporta.

E' stato cioe' creato un linguaggio, chiamato linguaggio simbolico, molto piu' vicino a quello da noi abitualmente usato, e che quindi non richiede lo sforzo mentale continuo di adattamento alla macchina.

E' evidente pero' che il lavoro di cui viene alleggerito l'uomo non puo' che essere trasferito all'elaboratore che viene dotato in questo caso di un particolare dispositivo che gli permette la traduzione di questo nuovo linguaggio nei simboli ad esso intelligibili.

Nei capitoli che seguono si trovano illustrate le istruzioni, i comandi cioe' che si possono impartire alla macchina, se ne elencano le caratteristiche, ed infine si espongono le norme relative all'utilizzazione dei nastri e dei programmi plurisequenziali .

6.2. La codificazione delle informazioni

Le informazioni - dati, risultati e istruzioni di programma - sono rappresentate per mezzo di caratteri alfanumerici e speciali, la cui codificazione varia secondo il supporto utilizzato per la loro registrazione.

Per i seguenti supporti : la memoria principale e i registri del calcolatore, i nastri ed i tamburi magnetici ed infine la banda perforata, ogni carattere e' rappresentato da un insieme di sette variabili binarie denominate "bit": sei bit servono per la rappresentazione del carattere vero e proprio, il settimo, che viene denominato bit di disparita', ser

LA CONFIGURAZIONE IN BIT DEI CARATTERI DELL'ELEA 9003

ZONA F e	0000	0001	0011	0111	0100	1000	1001	1011	1111	1100	0010	0101	0110	1010	1101	1110
0 0	0	1	2	3	4	5	6	7	8	9	~	+	#	/	−	÷
0 1	Ø	A	B	C	D	E	F	G	H	I	δ	=	α	%	,	γ
1 1	ε	J	K	L	M	N	O	P	Q	R	*	(	β	$	η	π
1 0	·	S	T	U	V	W	X	Y	Z	'	Θ	!	)	?	&	⊗

LA CONFIGURAZIONE DEI CARATTERI DELL'ELEA 9003 SU NASTRO PERFORATO

6 5 4 3 2 1

ZONA 6 5	0001	1001	1000	0100	1100	1010	0110	0101	0011	0010	0000	0111	1111	1011	1101	1110
	ZONA 4 3 2 1															
0 0	0	1	2	3	4	5	6	7	8	9	~	+	#	/	–	÷
0 1	ϕ	A	B	C	D	E	F	G	H	I	δ	=	α	%	,	γ
1 1	Ɛ	J	K	L	M	N	O	P	Q	R	*	(	β	$	η	π
1 0	·	S	T	U	V	W	X	Y	Z	’	⊖	!	)	?	&	⊗

ve per il controllo dell'esatta registrazione delle informazioni. Piu' precisamente il settimo bit e' 0 o 1 in modo che il numero dei bit 1 che compaiono in ogni carattere sia dispari. Su banda perforata non compare il settimo bit.

Tali bit sono materializzati :

- da posizioni di perforazione sulla banda perforata;

- da nuclei magnetizzabili nella memoria principale e nei registri del calcolatore;

- da aree di magnetizzazione nei nastri e tamburi magnetici.

I sei bit del codice permettono ovviamente di rappresentare 64 caratteri fra cifre decimali, lettere, punteggiatura, principali simboli matematici, e altri caratteri speciali, che solitamente vengono utilizzati per l'organizzazione delle informazioni.

Per "codice del calcolatore" si intende il codice usato per le memorie a nuclei magnetici e per i nastri e tamburi magnetici; per "codice di perforazione" il codice utilizzato nella perforazione della banda. Le lettere a, b, c, d, e, f, nel codice del calcolatore, ed i numeri 1, 2, 3, 4, 5, 6 nel codice di perforazione, indicano la posizione dei diversi bit nell'ambito del carattere rappresentato.

I singoli caratteri possono essere raggruppati con diversi criteri in modo da formare parole, elementi, blocchi, sequenze, ecc. La parola e' formata da uno o piu' caratteri e identifica una unita' di informazione : una quantita' numerica, un nome, una data, ecc.

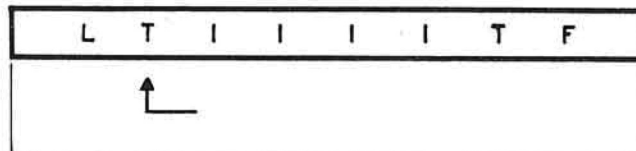
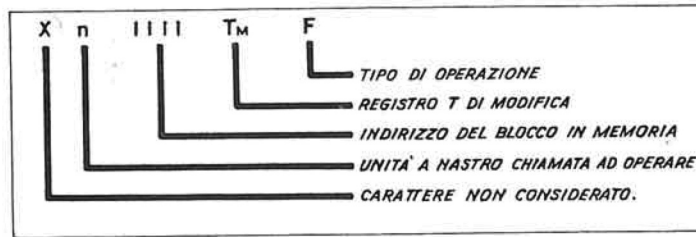
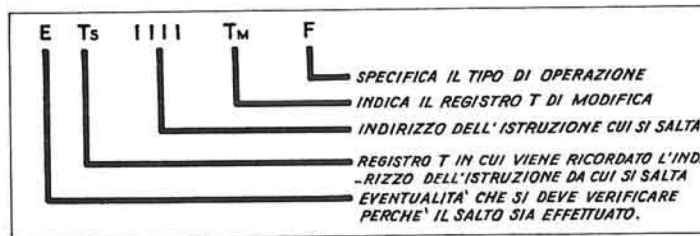
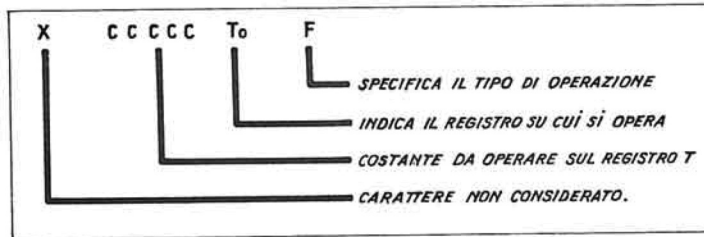
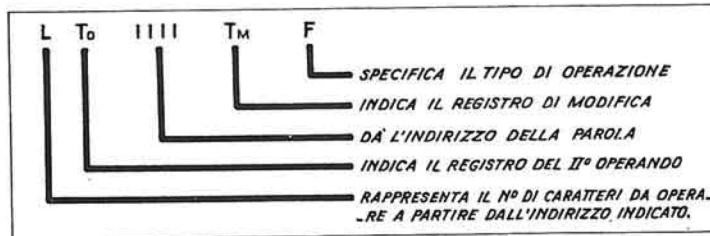
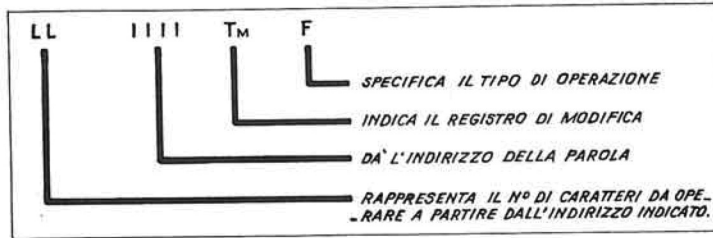
Durante l'elaborazione una parola puo' essere identificata mediante il suo indirizzo iniziale di memoria e la sua lunghezza oppure per mezzo del suo carattere estremo scelto a piacere tra i 64 possibili.

Un elemento e' composto da una o piu' parole e rappresenta, ad esempio, un fatto contabile o amministrativo, come un movimento bancario o di magazzino, ecc.

Un blocco e' composto da uno o piu' elementi. Una sequenza e' composta da uno o piu' blocchi.

6.3. La codificazione delle istruzioni

Una istruzione di programma e' codificata per mezzo di otto caratteri alfanumerici. Questi hanno normalmente la seguente funzione : il primo - nell'ordine in cui sono esaminati dall'unita' di governo - indica la funzione F, che definisce di quale tipo di istruzione si tratta; il secondo indica il registro T modificatore, il cui contenuto modifica l'indirizzo; i quattro seguenti contengono un indirizzo I ; gli ultimi due specificano la lunghezza L dell' informazione da trattare.

Vi sono pero' molte importanti eccezioni a questa regola : ad esempio, per tutte le istruzioni di salto l'ultimo carattere specifica il tipo di salto; le istruzioni interessanti memorie e registri T hanno la configurazione :

LA CONFIGURAZIONE DELLE ISTRUZIONI DELL'ELEA 9003

Qualunque sia il significato degli otto caratteri di una istruzione, la sua esecuzione si svolge in due fasi : una fase preparatoria ed una fase esecutiva.

Alcune istruzioni non comportano fase esecutiva.

La fase preparatoria ha una durata di nove periodi di cifra, che si indicano con i simboli da p0 a p8; i periodi da p1 a p8 sono utilizzati per la lettura e l'analisi dei caratteri dell'istruzione, e per la predisposizione degli organi interessati all'istruzione stessa.

In molti casi alcune posizioni dell'istruzione non hanno significato, e non sono neppure esaminate dall'unita' di governo, in queste posizioni si scrivera', per convenzione, la lettera X.

Esse possono contenere un carattere qualsiasi, e possono quindi essere utilizzate per la registrazione di costanti o altre informazioni.

Si e' visto che la posizione p2 dell'istruzione indica il registro T che deve intervenire a modificare l'istruzione stessa. Se non si desidera che questa venga modificata e' necessario indicare un registro T inesistente : cio' si ottiene registrando in p2 il carattere # (diesis), oppure il carattere ÷ (da ... a). E' questo un fatto generale : se in una istruzione una determinata posizione indica normalmente uno fra diversi organi della macchina, e non si vuole richiamare nessun organo di quel tipo, in essa si deve registrare un # (diesis).

Le posizioni da p3 a p6 sono normalmente riservate all'indirizzo, e precisamente :

p3 indica le unita' dell'indirizzo

p4 indica le decine dell'indirizzo

p5 indica le centinaia dell'indirizzo

p6 indica le migliaia dell'indirizzo.

Le unita' e decine sono sempre espresse mediante caratteri numerici; le centinaia e le migliaia invece possono essere espresse con caratteri sia numerici che alfabetici o speciali : cio' permette di indicare con solo 4 caratteri tutti gli indirizzi della memoria principale. La tabella N. 3 in appendice riporta la corrispondenza fra i caratteri registrati in p5 e p6 e le migliaia e centinaia.

Esempi :

L'indirizzo 12012 sara' indicato B012

L'indirizzo 31152 sara' indicato S152

L'indirizzo 42500 sara' indicato 2E00.

L'indirizzo che compare nell'istruzione e' quello della posizione di memoria in cui compare il segno o il carattere meno significativo della parola da operare. L'operazione inizia dall'indirizzo specificato e procede per indirizzi decrescenti. Cosi' ad esempio l'indirizzo della parola 546ABUL e' 1015.

		← - - - - - - - - - - - - - - - -							
Caratteri		5	4	6	A	B	U	L	
Posizioni di memoria	1008	1009	1010	1011	1012	1013	1014	1015	1016

CARATTERI RAPPRESENTANTI LE MIGLIAIA E LE CENTINAIA NEI DIVERSI GRUPPI DI MEMORIA

	DECINE MIGLIAIA	UNITÀ MIGLIAIA										UNITÀ CENTINAIA									
		0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9
40.000 POSIZIONI	0	0	1	2	3	4	5	6	7	8	9										
	1	ϕ	A	B	C	D	E	F	G	H	I	0	1	2	3	4	5	6	7	8	9
	2	Ɛ	J	K	L	M	N	O	P	Q	R										
	3	.	S	T	U	V	W	X	Y	Z	,										
		0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9
80.000 POSIZIONI	4	0	1	2	3	4	5	6	7	8	9										
	5	ϕ	A	B	C	D	E	F	G	H	I	ϕ	A	B	C	D	E	F	G	H	I
	6	Ɛ	J	K	L	M	N	O	P	Q	R										
	7	.	S	T	U	V	W	X	Y	Z	,										
		0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9
120.000 POSIZIONI	8	0	1	2	3	4	5	6	7	8	9										
	9	ϕ	A	B	C	D	E	F	G	H	I	Ɛ	J	K	L	M	N	O	P	Q	R
	10	Ɛ	J	K	L	M	N	O	P	Q	R										
	11	.	S	T	U	V	W	X	Y	Z	,										
		0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9
160.000 POSIZIONI	12	0	1	2	3	4	5	6	7	8	9										
	13	ϕ	A	B	C	D	E	F	G	H	I	.	S	T	U	V	W	X	Y	Z	,
	14	Ɛ	J	K	L	M	N	O	P	Q	R										
	15	.	S	T	U	V	W	X	Y	Z	,										

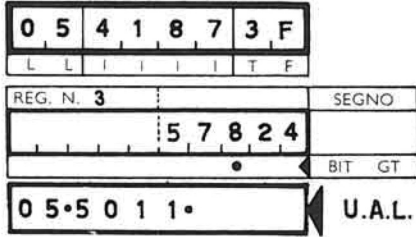
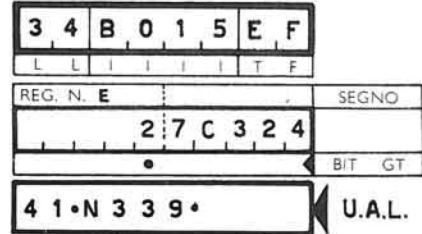
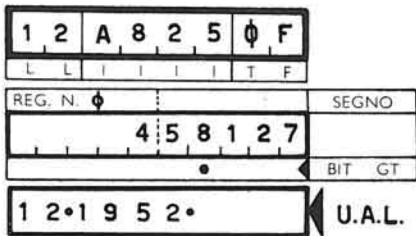
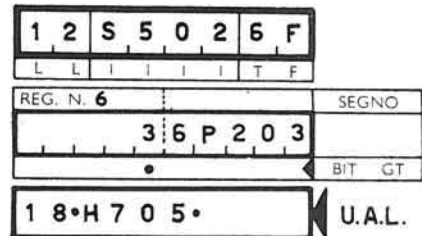
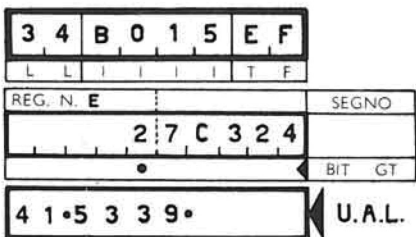
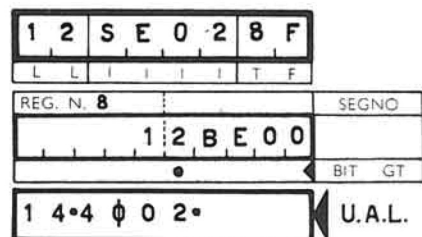
Tabella N. 3

6.4. Modifica automatica delle istruzioni

Per ogni istruzione e' indicato se essa e' modificabile automaticamente, nel qual caso il registro T modificatore e' specificato nella posizione p2.

L'operazione di modifica avviene nel modo seguente: i caratteri che si trovano in posizione p3 p4 p5 p6 p7 dell'istruzione vengono sommati ai caratteri contenuti nel T modificatore, dall'inizio di questo fino al bit gT per un massimo di cinque posizioni. Le posizioni p3, p4 e p5 dell'istruzione, cosi' come le corrispondenti posizioni 1ª, 2ª e 3ª del T modificatore, possono contenere solo cifre decimali; su queste posizioni la somma viene effettuata secondo le regole della aritmetica in base 10 e gli eventuali riporti passano dalla 1ª alla 2ª posizione, dalla 2ª alla 3ª, dalla 3ª alla 4ª. La posizione p6 dell'istruzione e la 4ª posizione del T modificatore possono invece contenere i caratteri alfabetici o speciali utilizzati per rappresentare le migliaia dell'indirizzo fino a 39.000 (da 40.000 a 79.000 , da 80.000 a 119.000, da 120.000 a 159.000 a seconda della capacita' della memoria principale). La addizione di questi due caratteri da' come risultato il carattere che rappresenta la somma delle migliaia indicate nei due addendi (tenendo conto dell'eventuale riporto dalla 3ª posizione di somma); oppure se viene superato il trentanovesimo migliaio, il carattere che rappresenta la stessa somma diminuito di quaranta. Non vi e' mai passaggio di riporto alla 5ª posizione di somma, ma di tale riporto si tiene conto sulla 4ª posizione stessa.

In altre parole, la somma sulla 4ª posizione si effettua secondo una aritmetica in base quaranta con soppressione del riporto; le cifre di tale aritmetica sono elencate nella tabella N. 3.

ESEMPI DI MODIFICA AUTOMATICA DELLE ISTRUZIONI

ES. 1 - 20'000 POSIZIONI

ES. 2 - 40'000 POSIZIONI

ES. 3 - 20'000 POSIZIONI

ES. 4 - 40'000 POSIZIONI

ES. 5 - 20'000 POSIZIONI

ES. 6 - 80'000 POSIZIONI

E' chiaro che tali regole permettono la modifica automatica degli indirizzi entro 40.000 posizioni di memoria circolare.

L'indicazione delle centinaia nei gruppi di memoria oltre le 40.000 posizioni si ha mediante caratteri alfabetici o speciali.

Dal carattere usato per le centinaia l'elaboratore riconosce il gruppo di appartenenza di un indirizzo.

Finalmente la 5ª posizione del T modificatore viene sommata, secondo le regole dell'aritmetica in base 10, alla posizione p7 dell'istruzione; tali posizioni possono quindi contenere solo cifre decimali. L'eventuale riporto viene sommato al carattere contenuto nella posizione p8 dell'istruzione.

La modifica automatica non esige alcun tempo supplementare di fase preparatoria; viene poi eseguita l'istruzione modificata (nella quale ovviamente l'indicazione del T modificatore non ha alcun interesse).

Il contenuto del T modificatore non viene alterato dalla modifica automatica dell'istruzione.

6.5. Lunghezza degli operandi nelle istruzioni interne.

Le istruzioni interne sottoindicate possono specificare la lunghezza dell'operando, o indicando con due caratteri numerici posti in p7 e p8 (LL) il numero dei caratteri costituenti l'operando, oppure indicando il carattere registrato in memoria dopo l'ultimo carattere dell'operando stesso.

Cio' puo' essere fatto in due modi diversi :

a) il carattere che segue l'ultimo carattere dell'operando e' ben determinato e conosciuto dal programmatore. Esso deve essere allora registrato nella posizione p8 mentre nella posizione p7 si deve porre :

   - Q : se si vuole che, terminata l'operazione , l'indirizzo del carattere chiave sia trasferito nel registro T9 ;

   - P : se non interessa che l'indirizzo del carattere chiave sia trasferito nelle prime 4 posizioni del registro T9.

b) l'ultimo carattere dell'operando e' seguito da un carattere alfabetico o speciale. La posizione p8 e' allora non interessante, e nella posizione p7 si deve porre :

   - Z : se si vuole che, terminata l'operazione , l'indirizzo del carattere chiave sia trasferito in T9 ;

   - Y : se non interessa che l'indirizzo del carattere chiave sia trasferito nelle prime 4 posizioni del registro T9.

In entrambi i casi l'operazione prosegue fino a che viene letto un carattere non numerico.

Il carattere chiave letto in memoria e' sempre rigenerato nella posizione di memoria in cui si trova.

Agli effetti del calcolo il carattere chiave e' considerato uno zero di memoria, o piu' in generale un numero non significativo. Esso indica sempre la fine della parola in memoria.

Se il carattere chiave ricercato si trova all'indirizzo IIII, specificato nell'istruzione interessata, la macchina si arresta su tale posizione ed in T9 viene registrato detto indirizzo.

Cio' non avviene se il carattere chiave e' un segno "piu' " o "meno" .

Le istruzioni che operano secondo le modalita' a) e b) precedentemente indicate sono :

MA, AM, AoM, +MA, +AM, -MA, CMA ;

MEM, +MM, -MM, CMM ;

Y, +X, -X ;

TOL.

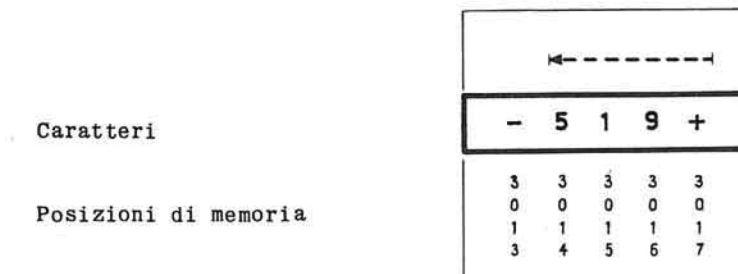
Le istruzioni che operano soltanto nel modo b) sono :

MT, TM, ToM, +MT, -MT, CMT ;

XLD, XLN, +LD.

Eventuali eccezioni a queste regole sono riportate nella descrizione delle singole istruzioni.

Esempio :

Nel caso che si voglia trasferire in accumulatore mediante un'istruzione MA la parola 519+ registrata in memoria come in figura :

la parte lunghezza e indirizzo dell'istruzione puo' essere espressa :

L	L	I	I	I	I
0	4	3	0	1	7
-	Q	3	0	1	7
-	P	3	0	1	7
X	Z	3	0	1	7
X	Y	3	0	1	7

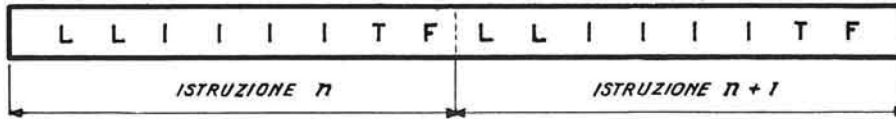
6.6. Registrazione del programma

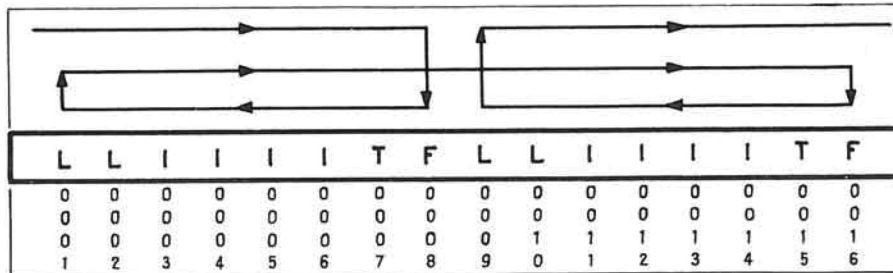
Perche' possano essere esaminate ed eseguite dalla unita' centrale, le istruzioni di programma devono essere registrate nella memoria principale.

Il modo normale di leggere una informazione in memoria principale e' di leggerla - carattere per carattere - per indirizzi decrescenti; l'indirizzo dell'informazione in memoria e' pertanto l' indirizzo del suo estremo carattere di destra.

Conseguentemente, le istruzioni in memoria principale sono registrate come e' indicato dalla figura; il carattere F, che sara' esaminato per primo dall'unita' di governo, occupa la posizione di memoria ad indirizzo piu' elevato; il carattere T la posizione di indirizzo immediatamente minore, e cosi' di seguito.

Se dopo l'istruzione n deve essere eseguita la istruzione n+1, questa dovra' essere registrata in memo-

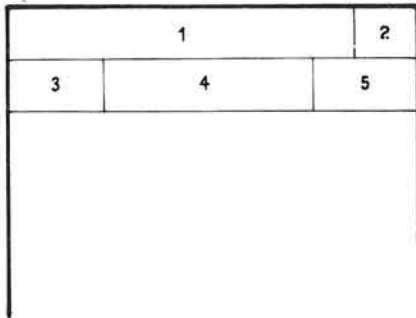
ria alla destra della precedente.

Il programma puo' essere registrato in una zona qualsiasi della memoria; questa puo' infatti contenere indifferentemente dati da elaborare, risultati o istruzioni di programma, in ogni sua posizione. Supponiamo di registrare il programma nelle prime posizioni di memoria, a partire dall'indirizzo 0001: il carattere F della prima istruzione avra' quindi lo indirizzo 0008, il carattere F della seconda istruzione l'indirizzo 0016, e cosi' via.

Da quanto si e' detto precedentemente, risulta chiaro che, perche' l'organo di governo esegua una istruzione deve rivolgersi all'indirizzo del carattere F di quella istruzione (dunque nel caso anzidetto, successivamente agli indirizzi 0008, 0016, ecc.). Tale indirizzo si dice brevemente indirizzo dell'istruzione.

Ad ogni istruzione e' stato dato anche un codice simbolico piu' facile da ricordarsi mnemonicamente e quindi piu' comodo nella prima stesura di un programma. Cosi' ad esempio l'istruzione per confrontare fra di loro due zone di memoria che ha come codice di funzione il carattere "/" viene indicata con il codice simbolico CMM.

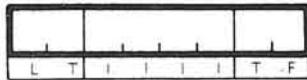
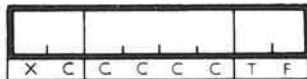
Il programma normalmente viene svolto eseguendo successivamente le istruzioni una dopo l'altra nell'ordine in cui sono registrate in memoria a partire da quella indicata all'avvio sul quadro di controllo; speciali organi di governo contengono l' indirizzo dell'istruzione. Si possono svolgere pero' contemporaneamente tre diversi programmi sequenziali come sara' meglio specificato in seguito.

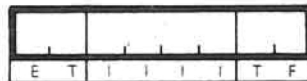
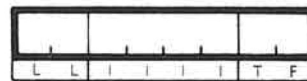
Questo modo di procedere sequenzialmente puo' essere variato mediante apposite istruzioni (istruzioni di "salto") che permettono di eseguire sequenze diverse sia sistematicamente sia in funzione di eventi rilevati durante l'elaborazione, sia a seguito di condizioni impostate direttamente sul quadro di controllo. Vi sono 90 diversi tipi di istruzione di cui 38 di salto. Le istruzioni di "salto" specificano tra l'altro l'indirizzo della prossima istruzione da eseguire.

LEGENDA

1		2
3	4	5

Schema istruzioni :

1 - *Funzione*
2 - *codice simbolico*
3 - *organi impegnati durante la fase esecutiva*
4 - *configurazione*
5 - *tempo, in periodi di cifra, necessario all'esecuzione dell'istruzione (fasi α e β).*

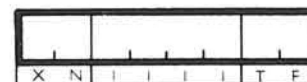
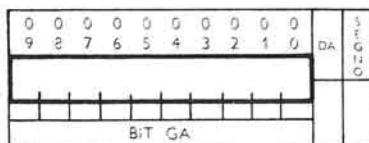
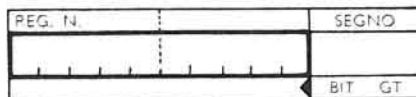
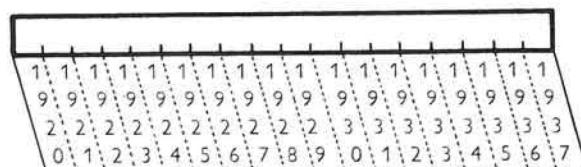
Configurazioni istruzioni

L T I I I I T F — *memoria - registri*

X C C C C C T F — *costante - registri*

E T I I I I T F — *salti*

L L I I I I T F — *memoria - accumulatore*
memoria - memoria

X N I I I I T F — *nastri*

0 0 0 0 0 0 0 0 0 0
9 8 7 6 5 4 3 2 1 0 DA SEGNO
BIT GA — *Accumulatore*

REG. N. SEGNO
BIT GT — *Registri T*

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7

Memoria :
contenuto, indirizzi

CAP. 7° : ISTRUZIONI RIGUARDANTI L'UNITA' CENTRALE

7.1. Istruzioni memoria-accumulatore

Queste istruzioni interessano la memoria principale e l'accumulatore; nel loro svolgimento impegnano la unita' aritmetica e logica ed il governo dell'elaboratore.

Sia durante la fase preparatoria che in quella esecutiva il canale interno risulta impegnato; a questo gruppo di istruzioni potranno percio' sovrapporsi operazioni che impegnano il canale esterno ed il governo unita' a nastro.

Queste istruzioni sono normalmente utilizzate :

a) per trasferimenti da memoria verso accumulatore e viceversa di parole segnate o non segnate di lunghezza conosciuta o sconosciuta, non superiore a 100 caratteri, hanno questa funzione le istruzioni :

MA, AM, AoM

b) per operazioni aritmetiche o algebriche su numeri di lunghezza variabile da 1 a 100 caratteri; hanno questa funzione le istruzioni :

+ MA, - MA, ÷ AM

c) per confronti tra valori assoluti, numeri algebrici o parole alfanumeriche, di lunghezza non superiore a 100 caratteri; ha questa funzione l'istruzione :

CMA

d) per determinare la lunghezza di una parola contenuta in accumulatore, escluso l'eventuale segno

algebrico.
In quest'ultimo caso e' necessario ricorrere a particolari accorgimenti di programmazione.

Sistema a) Sia registrata mediante istruzione FAM la lunghezza del numero segnato contenuto in accumulatore, nelle posizioni p7 e p8 di un'istruzione AM o AoM.
L'opportuna modifica di LL e' ottenibile mediante il registro modificatore indicato in p2 della stessa istruzione.
E' sufficiente infatti sommare al contenuto di questo registro il valore 10.000 perche' il bit gT passi in quinta posizione e la cifra piu' significativa incrementi di una unita' la lunghezza espressa mediante l'istruzione FAM.

Sistema b) Si provvede, mediante una istruzione CT, a registrare all'inizio del ciclo di elaborazione comprendente l'istruzione FAM il valore 1 in un determinato registro.
Si invia quindi mediante l'istruzione FAM la lunghezza del numero segnato contenuto in accumulatore sulle posizioni p3 e p4 di una + CT che permette di sommare detta lunghezza all'unita' preesistente nel registro precedentemente operato.
Si ottiene in questo modo nel registro la lunghezza esatta del numero algebrico (valore assoluto + segno) che puo' essere trasferita mediante TM nell'istruzione interessata all'uscita del dato dall'accumulatore.

Dispone accumulatore	DA

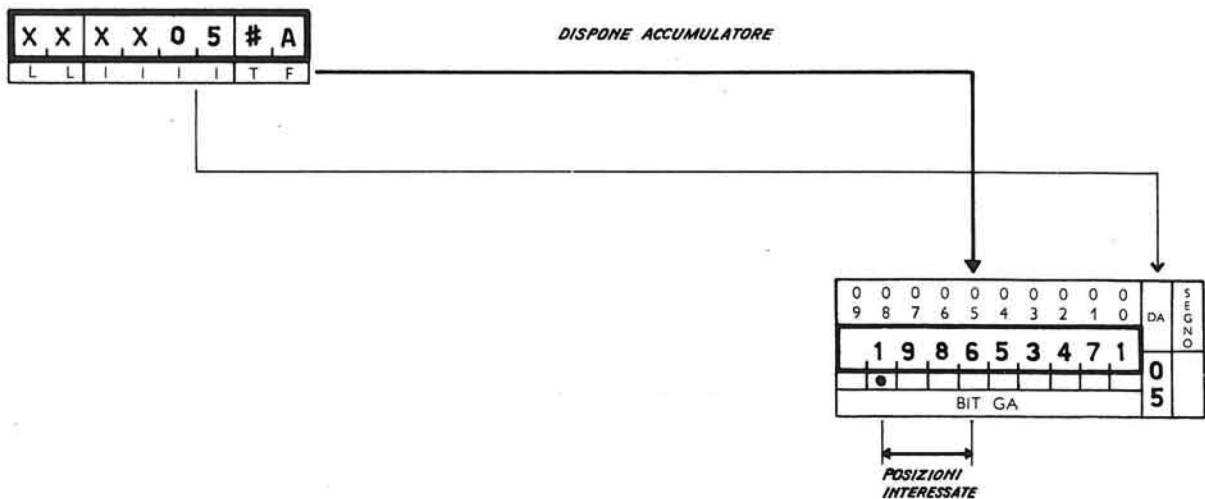
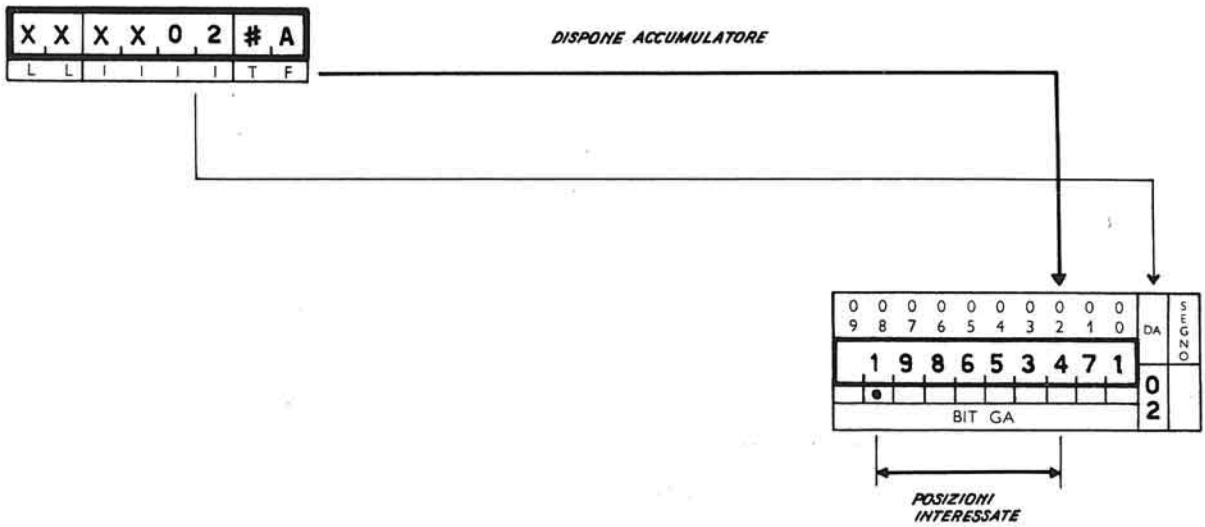
interno	X X X X I I T_m A (010001)	10

X X X X : posizioni non utilizzate

I I : indirizzo di accumulatore

T_m : il registro T_m puo' modificare l'indirizzo II

A : fissa l'inizio dell'accumulatore alla IIesima posizione

a) Il contenuto dell'accumulatore rimane immutato.

Trasferisce da memoria ad accumulatore			MA
interno	L L I I I I T_m F (011001)		10 + 1

L L : lunghezza dell'operando da trasferire in accumulatore

I I I I : indirizzo di memoria dell'operando che deve essere trasferito

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

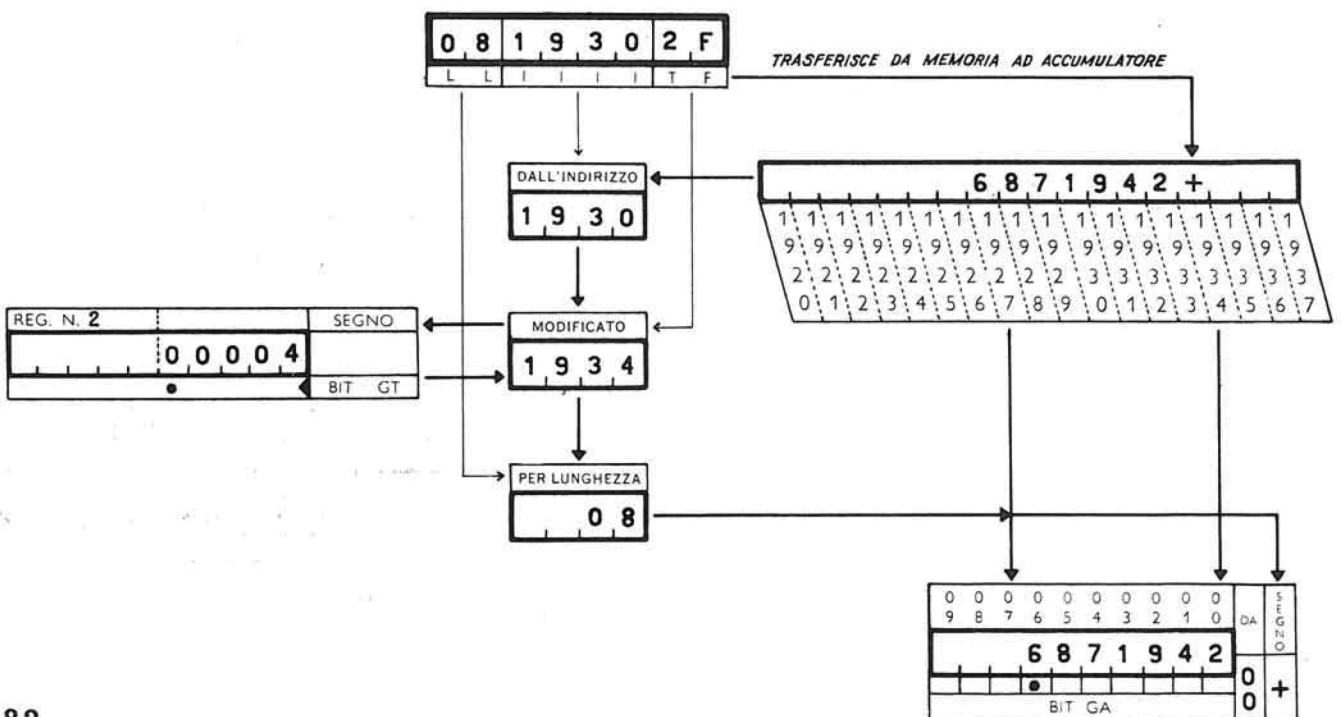
F : trasferisce dalla memoria a partire dall'indirizzo IIII per lunghezza LL e per indirizzi decrescenti all'accumulatore a partire da DA

a) La memoria non resta azzerata.

b) I caratteri trasferiti si sostituiscono a quelli precedentemente contenuti nell'accumulatore.

c) In corrispondenza dell'ultimo carattere trasferito viene posto un bit gA.

d) Se all'indirizzo IIII si trova un segno questo viene registrato nell'apposito registro del segno e l'informazione viene trasferita nell'accumulatore, da DA e per una lunghezza LL-1 a partire dall'indirizzo di memoria IIII-1.

e) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave; in accumulatore, invece del carattere chiave, viene trasferito uno zero, ed il bit gA e' posizionato sotto quest'ultimo.

Trasferisce da accumulatore a memoria senza azzeramento		AM
interno	L L I I I I T_m G (011011)	10 + 1

L L : lunghezza dell'operando da trasferire dall'accumulatore

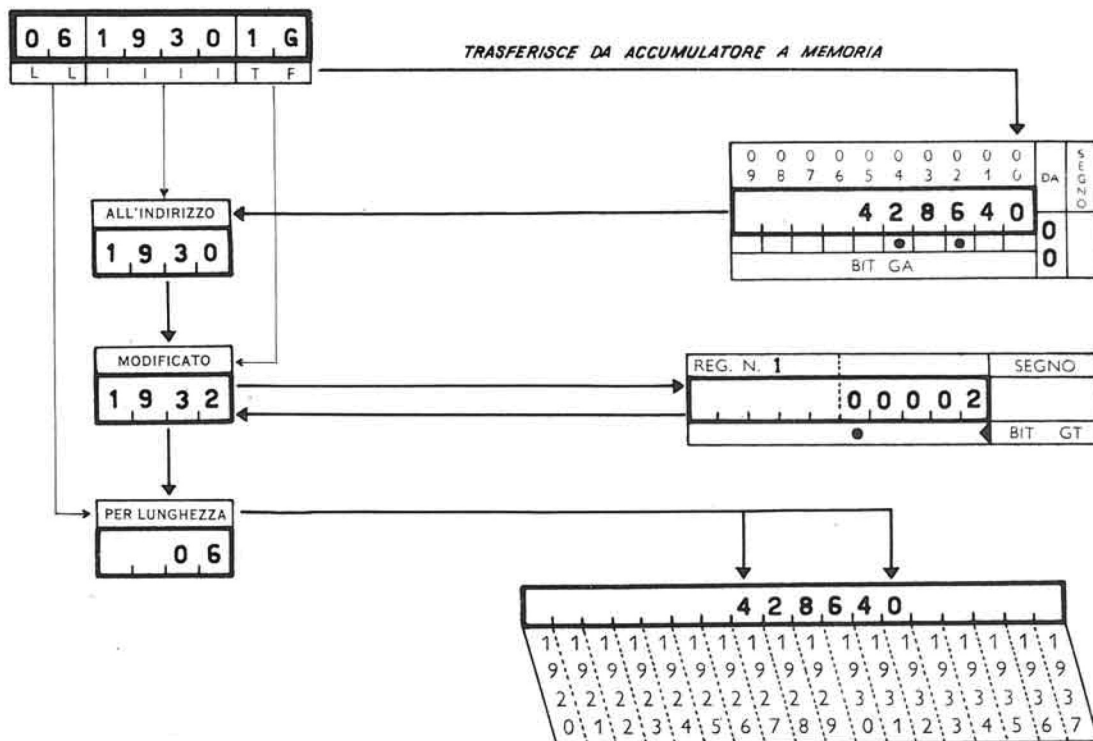
I I I I : indirizzo di memoria a cui l'operando deve essere trasferito

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

G : trasferisce dall'accumulatore a partire da DA, alla memoria a partire dall'indirizzo IIII per lunghezza LL e per indirizzi decrescenti

a) Il contenuto dell'accumulatore e del registro del segno, le posizioni del DA e del bit gA restano invariate.

b) Se la parola contenuta in accumulatore e' segnata, il segno viene registrato in memoria all'indirizzo IIII e i caratteri contenuti in accumulatore, a partire da DA, sono trasferiti in memoria a partire dall'indirizzo IIII-1 e per lunghezza LL-1.

c) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave : quest'ultimo deve essere contenuto in memoria.

Trasferisce da accumulatore a memoria con azzeramento			AoM
interno	L L IIII T_m B (010011)		10 + 1

L L : lunghezza dell'operando da trasferire dall'accumulatore

I I I I : indirizzo di memoria a cui l'operando deve essere trasferito

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

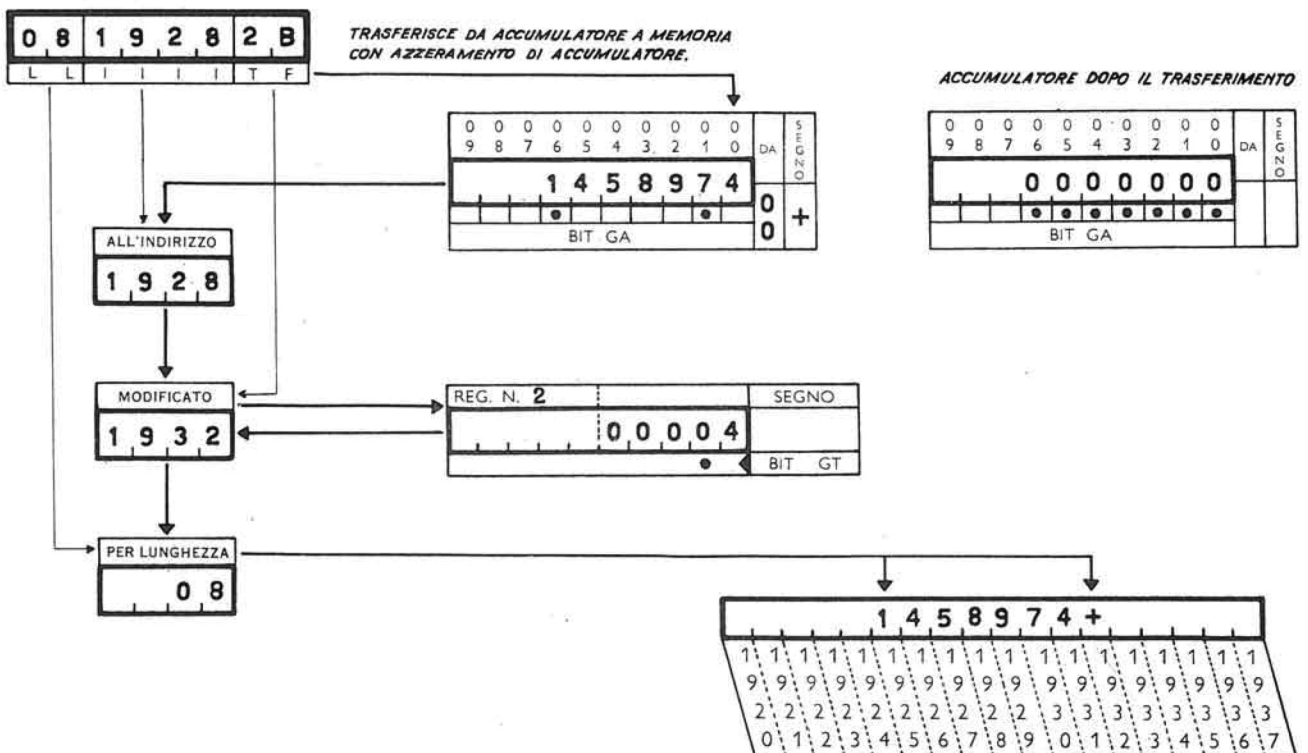
B : trasferisce dall'accumulatore a partire da DA, alla memoria a partire dall'indirizzo IIII per lunghezza LL e per indirizzi decrescenti

a) Le posizioni di accumulatore interessate al trasferimento vengono azzerate.

b) In ognuna di queste posizioni e' posto un bit gA.

c) Il registro del segno passa allo stato "non segnato, in vera grandezza".

d) Qualora nelle posizioni p7 e p8 fosse registrato il carattere # (diesis) l'esecuzione di una tale istruzione provoca l'azzeramento totale della memoria principale.

e) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave : quest'ultimo deve essere contenuto in memoria.

Somma memoria ad accumulatore, risultato in accumulatore		+ MA
interno	L L IIII T_m C (010111)	10 + 1

L L : lunghezza dell'operando registrato in memoria

I I I I : indirizzo di memoria dello stesso operando

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

C : somma il contenuto della memoria a partire dall'indirizzo IIII per lunghezza LL e per indirizzi decrescenti al contenuto dell'accumulatore compreso tra DA e bit gA

a) Il risultato si trova in accumulatore a partire da DA.

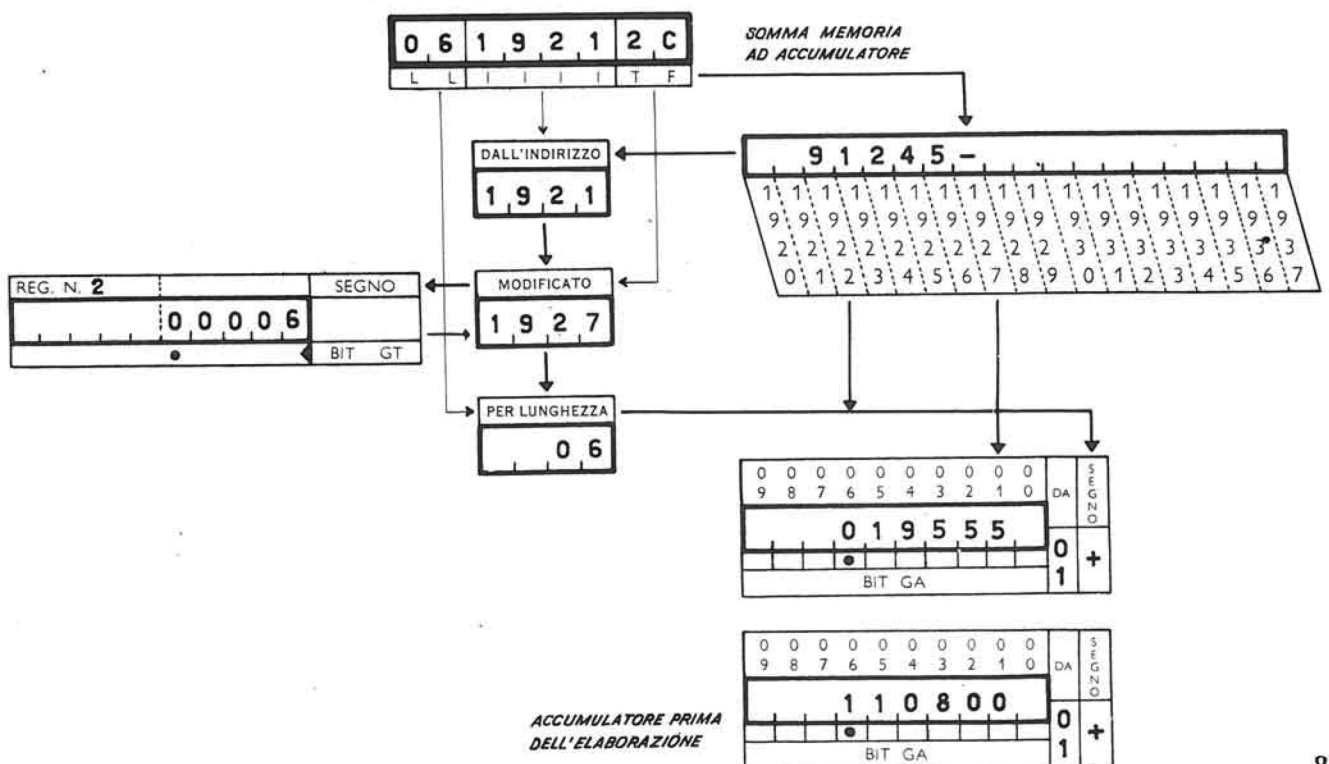
b) Il contenuto delle posizioni di memoria interessate dall'istruzione rimane inalterato.

c) L'operazione si effettua di regola sommando caratteri numerici.

d) Qualora in uno o in entrambi gli addendi siano presenti caratteri alfabetici, questi vengono sommati secondo la logica aritmetica esposta nel manuale.

e) E' sufficiente che uno dei due operandi sia segnato perche' la operazione risulti algebrica.

f) Se uno solo degli operandi e' segnato l'altro e' considerato positivo.

g) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave.

Somma accumulatore a memoria, risultato in memoria						+AM
interno	L L	I I I I	T_m	γ	(011110)	10 +1

L L : lunghezza dell'operando registrato in memoria e lunghezza massima dell'operando registrato in accumulatore

I I I I : indirizzo dell'operando registrato in memoria

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

γ : somma il contenuto dell'accumulatore compreso tra DA e bit gA al contenuto della memoria a partire dall'indirizzo IIII per lunghezza LL e per indirizzi decrescenti

a) Se LL e' maggiore del numero delle posizioni di accumulatore comprese tra DA e bit gA, le cifre oltre il bit gA non vengono operate.

b) Se LL e' minore del numero delle posizioni di accumulatore comprese tra DA e bit gA, le cifre oltre la lunghezza LL non vengono operate, e si ha indicazione di overflow.

c) Il risultato si forma in memoria a partire dall'indirizzo IIII.

d) Il contenuto dell'accumulatore rimane invariato.

e) Non si ha passaggio di riporto oltre la lunghezza specificata in LL.

f) Si ha indicazione di overflow nel caso in cui si dovrebbe avere riporto oltre la lunghezza indicata.

g) L'operazione puo' essere eseguita anche tra parole segnate con le seguenti limitazioni :

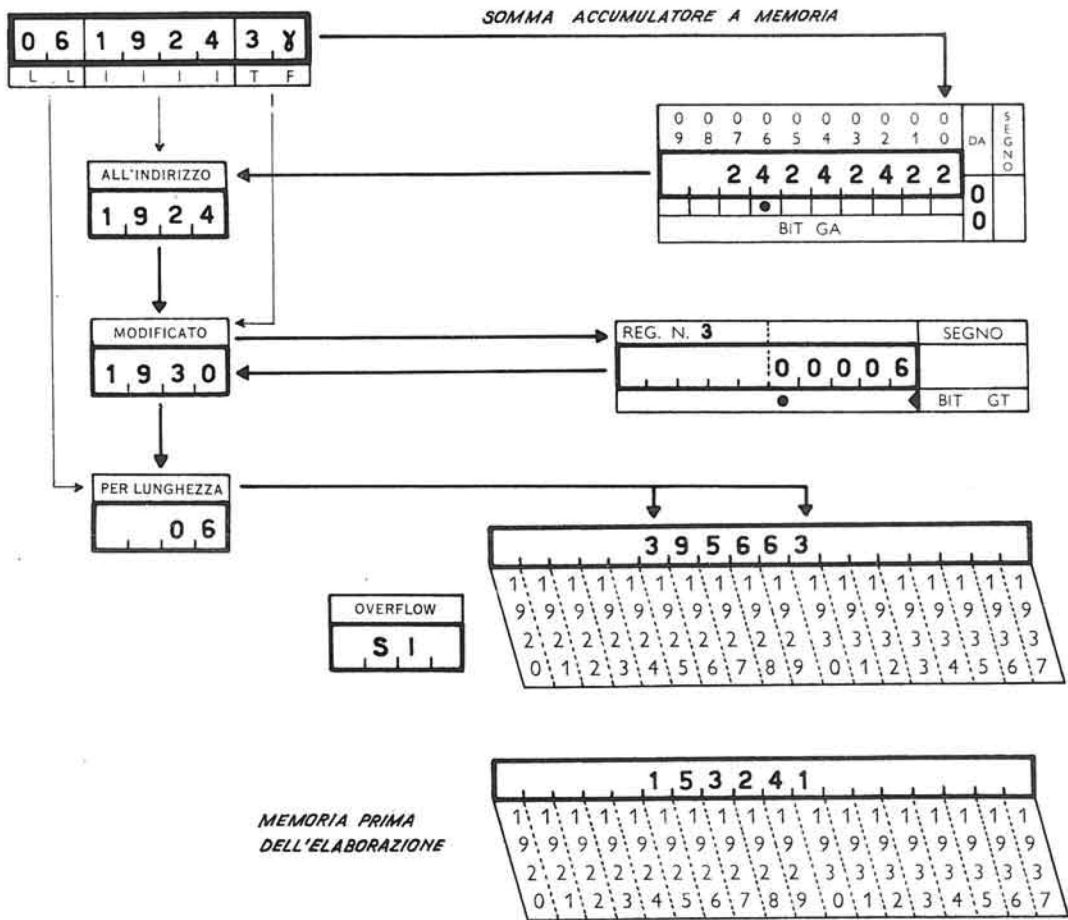
1 - del segno in accumulatore si tiene conto solo per stabilire se il contenuto di accumulatore va sommato o sottratto al contenuto delle posizioni di memoria;

2 - se la memoria e' segnata, l'indirizzo IIII deve essere quello del segno;

3 - il segno di memoria non viene preso in considerazione e pertanto rimane immutato;

4 - in caso di sottrazione, se il contenuto dell'accumulatore e' > di quello della memoria il risultato e' in complemento e si ha indicazione di overflow.

h) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave.

SOMMA ACCUMULATORE A MEMORIA
0 6 1 9 2 4 3 γ
L L I I I I T F
ALL'INDIRIZZO
1 9 2 4
DA
SEGNO
2 4 2 4 2 4 2 2
0
0
BIT GA
MODIFICATO
1 9 3 0
REG. N. 3
SEGNO
0 0 0 0 6
BIT GT
PER LUNGHEZZA
0 6
3 9 5 6 6 3
OVERFLOW
S I
1 5 3 2 4 1
MEMORIA PRIMA DELL'ELABORAZIONE

Sottrae memoria da accumulatore, risultato in accumul.		MA
interno	L L IIII T_m H (011111)	10 +1

L L : lunghezza dell'operando registrato in memoria

I I I I : indirizzo di memoria dello stesso operando

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

H : sottrae al contenuto dell'accumulatore compreso tra DA e bit gA il contenuto della memoria a partire dall'indirizzo IIII per lunghezza LL e per indirizzi decrescenti

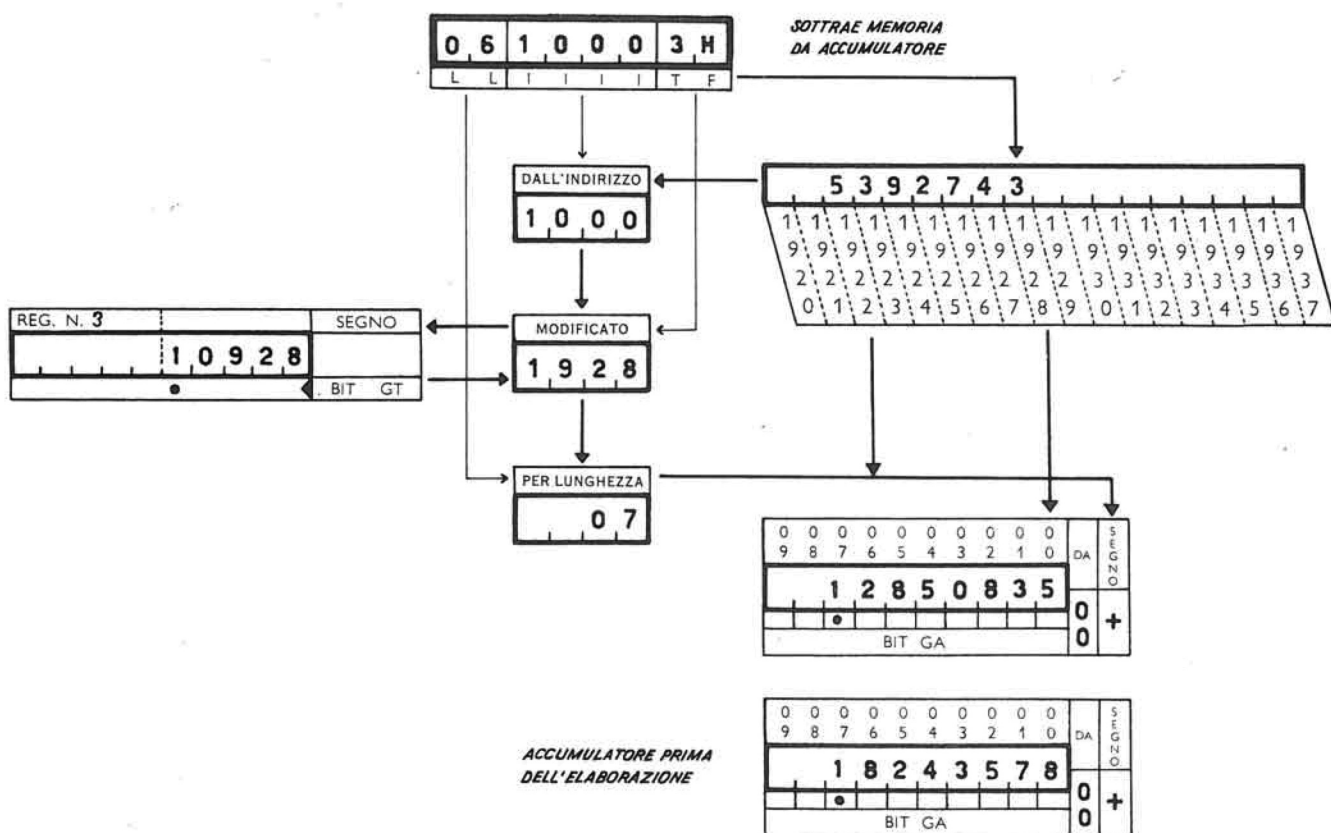
a) Il risultato si trova in accumulatore a partire da DA.

b) Il contenuto delle posizioni di memoria interessate dall'istruzione rimane inalterato.

c) L'operazione si effettua sommando al minuendo il complemento del sottraendo.

d) E' sufficiente che uno dei due operandi sia segnato perché l'operazione risulti algebrica.

e) Se uno solo degli operandi e' segnato l'altro e' considerato positivo.

f) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave.

Confronta memoria con accumulatore						CMA
interno	L L	I I I I	T_m	E	(011000)	10 + 1

L L : lunghezza del termine di confronto registrato in memoria

I I I I : indirizzo di memoria dello stesso operando

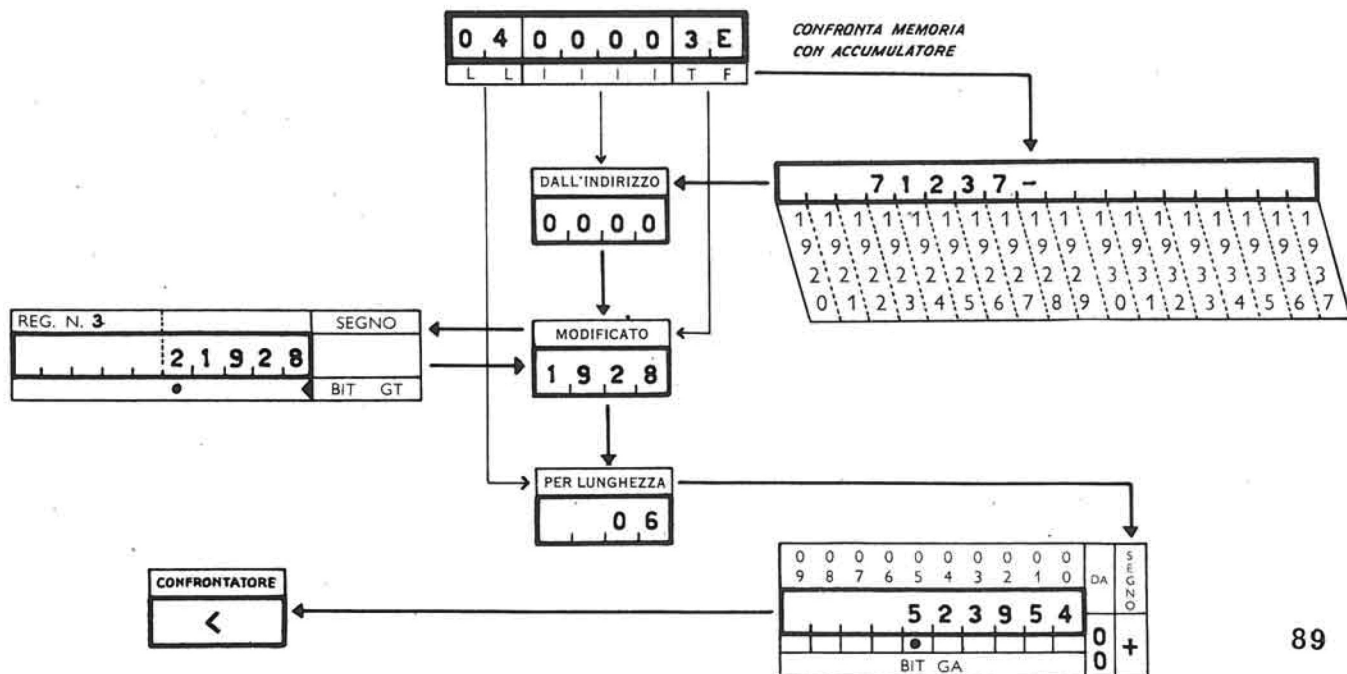
T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

E : confronta il contenuto della memoria a partire da IIII per lunghezza LL e per indirizzi decrescenti con il contenuto dell'accumulatore compreso tra DA e bit gA

a) Il confronto e' eseguito per tutti i caratteri anche non numerici, in base all'ordine gerarchico dei caratteri (vedi elenco dei caratteri).

b) Nel confronto si tiene conto dei segni se questi sono :
- per l'informazione contenuta in memoria, il primo carattere chiamato;
- per quella contenuta in accumulatore, il carattere contenuto nel registro del segno;
- se i caratteri + e − sono contenuti nell'informazione diversamente da come detto prima, risulta − > + ;
- se delle due parole da confrontare una sola e' segnata, quella non segnata e' considerata positiva.

c) Il risultato del confronto e' ricordato dall'unita' di governo e puo' quindi servire a condizionare il proseguimento dell'elaborazione fino a che non viene effettuato un nuovo confronto (CMA, CMT, CMM, CCT).

d) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave : quest'ultimo deve essere contenuto in memoria.

Fine accumulatore in memoria			FAM
interno	0 1 I I I I T_m δ (010010)		12 + 1

0 1 : caratteri che determinano il genere di lunghezza ricercata

I I I I : indirizzo a cui viene portata la lunghezza richiesta

T_m : il registro T_m puo' modificare l'indirizzo IIII

δ : trasferisce all'indirizzo IIII e per indirizzi decrescenti il numero di cifre contenute in accumulatore da DA a bit gA.

a) Se in p7 e p8 vengono registrati i caratteri 0 1 (zero, uno), all'indirizzo IIII e' trasferito il numero delle cifre significative comprese tra DA e bit gA.

b) Se in p7 e p8 vengono registrati i caratteri . 0 (punto zero), all' indirizzo IIII e' trasferito il numero delle cifre significative e non significative comprese tra DA e bit gA.

c) La lunghezza richiesta viene registrata sempre mediante 2 cifre.

d) Se richiedendo il numero di cifre significative (caso "a") non ne risultasse alcuna, vengono trasferiti all'indirizzo IIII e IIII - 1, i caratteri 0 1 (zero, uno).

Esempio della nota d)

0 1 1 9 2 8 # S
L L I I I I T F
TRASFERISCE IN MEMORIA Nº CIFRE SIGNIFICATIVE DI ACCUMULATORE.
ALL'INDIRIZZO
1 9 2 8
0 0 0 0 0 0 0 0 0 0
9 8 7 6 5 4 3 2 1 0
DA
SEGNO
0 0 4 5 2 9 1
0
1
BIT GA
MODIFICATO
#
NO
REG. N.
SEGNO
BIT GT
0 5 1 4 0 0 2 G

• 0 1 9 2 8 # S
L L I I I I T F
TRASFERISCE IN MEMORIA Nº CIFRE SIGNIFICATIVE E NON SIGNIFICATIVE DI ACCUMULATORE.
ALL'INDIRIZZO
1 9 2 8
0 0 0 0 0 0 0 0 0 0
9 8 7 6 5 4 3 2 1 0
DA
SEGNO
0 0 4 5 2 9 1
0
1
+
BIT GA
MODIFICATO
#
NO
REG. N.
SEGNO
BIT GT
0 7 1 4 0 0 2 G

SCHEMA RIASSUNTIVO	Istruzioni Memoria Accumulatore

Trasferimento	:	MA	AM	AoM
Operazioni aritmetiche	:	+MA	+AM	-MA
Confronto	:	CMA		
Determinazione lunghezza	:	FAM		

Particolarita' dell'Accumulatore

Capacita' : 100 posizioni.

DA : fissa l'indirizzo dell'accumulatore.

Bit gA : segnale fine-informazione in accumulatore.

Regole :
a) Non viene spostato se l'operando in memoria ha lunghezza uguale al numero di posizioni comprese tra DA e bit gA ;
b) Viene posto in corrispondenza dell'ultima cifra trasferita in accumulatore;
c) Viene posto in corrispondenza della cifra piu' significativa del risultato.

Registro del segno : da indicazione del segno e della configurazione del risultato in accumulatore.

Per le operazioni aritmetiche : la regola di complementazione e' la seguente: la macchina esegue la complementazione se il numero di segni "meno" che risulta dall'esame del segno dell'operazione, del segno dell'operando in memoria, del segno dell'operando in accumulatore e del segno del moltiplicatore, e' dispari.
Per i segni e' prevista l'inversione del segno dell'operando complementato; sono poi valide le regole algebriche sui segni.

Per i trasferimenti : il registro indichera' :
- non segnato in vera grandezza dopo una istruzione AoM o una MA che operi su di un numero non segnato;
- segnato in vera grandezza dopo una istruzione MA che operi su di un numero segnato;
- segnato in complemento : il dato, con una istruzione tipo AM, viene trasferito in vera grandezza col segno effettivo;
- non segnato in complemento : il dato, con un'istruzione tipo AM, viene trasferito tale e quale.

CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE	CARATTERE DI FUNZIONE
DA	interno	10	X X X X I I T_m	A
MA	"	10 + 1	L L I I I I T_m	F
AM	"	10 + 1	L L I I I I T_m	G
AoM	"	10 + 1	L L I I I I T_m	B
+MA	"	10 + 1	L L I I I I T_m	C
+AM	"	10 + 1	L L I I I I T_m	γ
-MA	"	10 + 1	L L I I I I T_m	H
CMA	"	10 + 1	L L I I I I T_m	E
FAM	"	12 + 1	0 1 I I I I T_m	δ
FAM	"	12 + 1	. 0 I I I I T_m	δ

7.2. Istruzioni memoria-memoria

Queste istruzioni interessano soltanto la memoria principale; nel loro svolgimento impegnano l'unita' aritmetica e logica ed il governo dell'elaboratore.

La fase preparatoria di una istruzione di questo gruppo, come per gli altri tipi d'istruzione, impegna il solo canale interno, la fase esecutiva invece impegna ambedue i canali.

A questo gruppo di istruzioni possono percio' sovrapporsi operazioni eseguibili mediante il solo governo unita' a nastro.

Esse sono normalmente utilizzate :

a) per trasferimenti, da una zona di memoria ad una altra, di un numero variabile da 1 ad n-2 caratteri (n = numero totale delle posizioni di memoria); cio' si ottiene mediante le istruzioni :

PUM MEM

b) operazioni aritmetiche su parole di lunghezza variabile da 1 a n/2 caratteri; hanno queste funzioni le istruzioni :

PUM + MM , PUM - MM

c) confronti tra parole alfanumeriche oppure tra numeri entrambi segnati, di lunghezza variabile da 1 a n/2 caratteri; hanno questa funzione le istruzioni :

PUM CMM

Per ottenere l'azzeramento della memoria principale per un determinato numero di posizioni e' sufficiente non indicare alcun indirizzo nelle posizioni da p3 a p6 dell'istruzione PUM riferita ad una MEM.

Prepara uscita memoria		PUM
interno	L L IIII - T_m ÷ (001110)	10

L L : lunghezza dell'operando

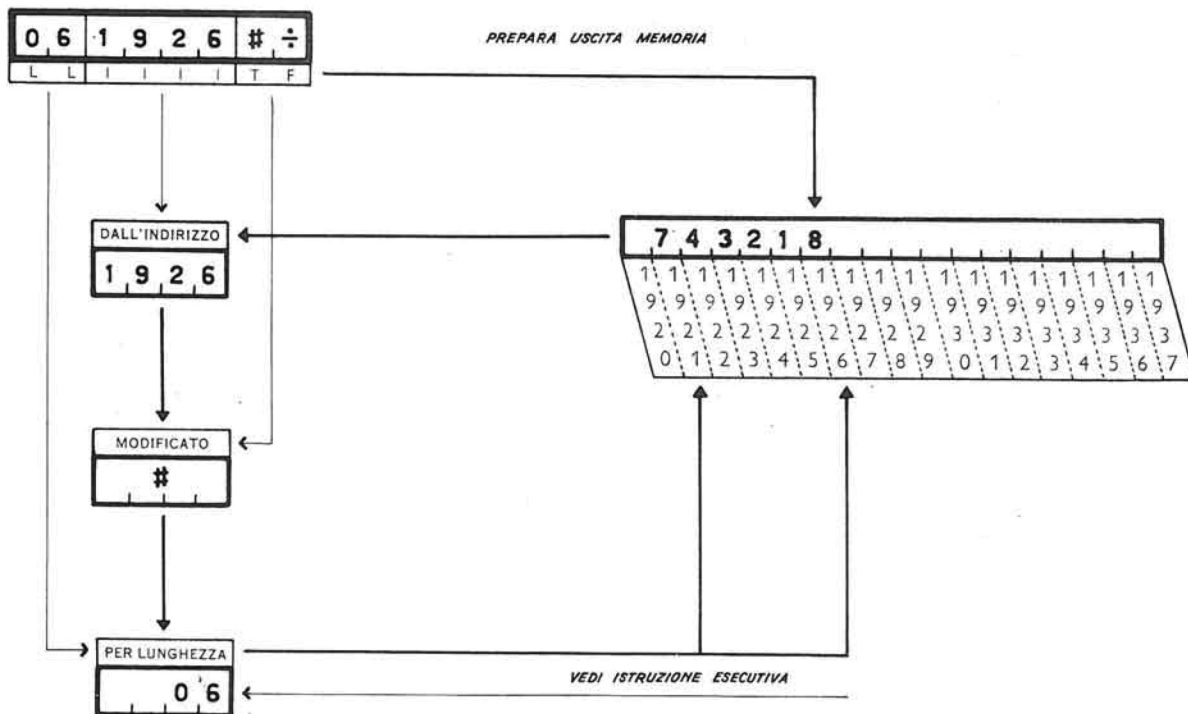
I I I I : indirizzo dell'operando in memoria

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

÷ : prepara l'uscita da memoria di un operando.

a) Se in p7 e in p8 e' registrato il carattere # (diesis) la lunghezza dovra' essere indicata dall'istruzione MEM, CMM, + MM, - MM che segue.

b) Se in p7 e p8 e' indicata la lunghezza, nella istruzione MEM, CMM, + MM o - MM che segue, in luogo della lunghezza dovranno essere registrati due zeri (00).

c) L'istruzione PUM deve essere immediatamente seguita nel programma dalla istruzione esecutiva.

Trasferisce da memoria a memoria						MEM
esterno ed interno	L L	I I I I	T_m	1	(000001)	10 + 1

L L : lunghezza dell'operando indicato nella PUM

I I I I : indirizzo di memoria a cui l'operando va trasferito

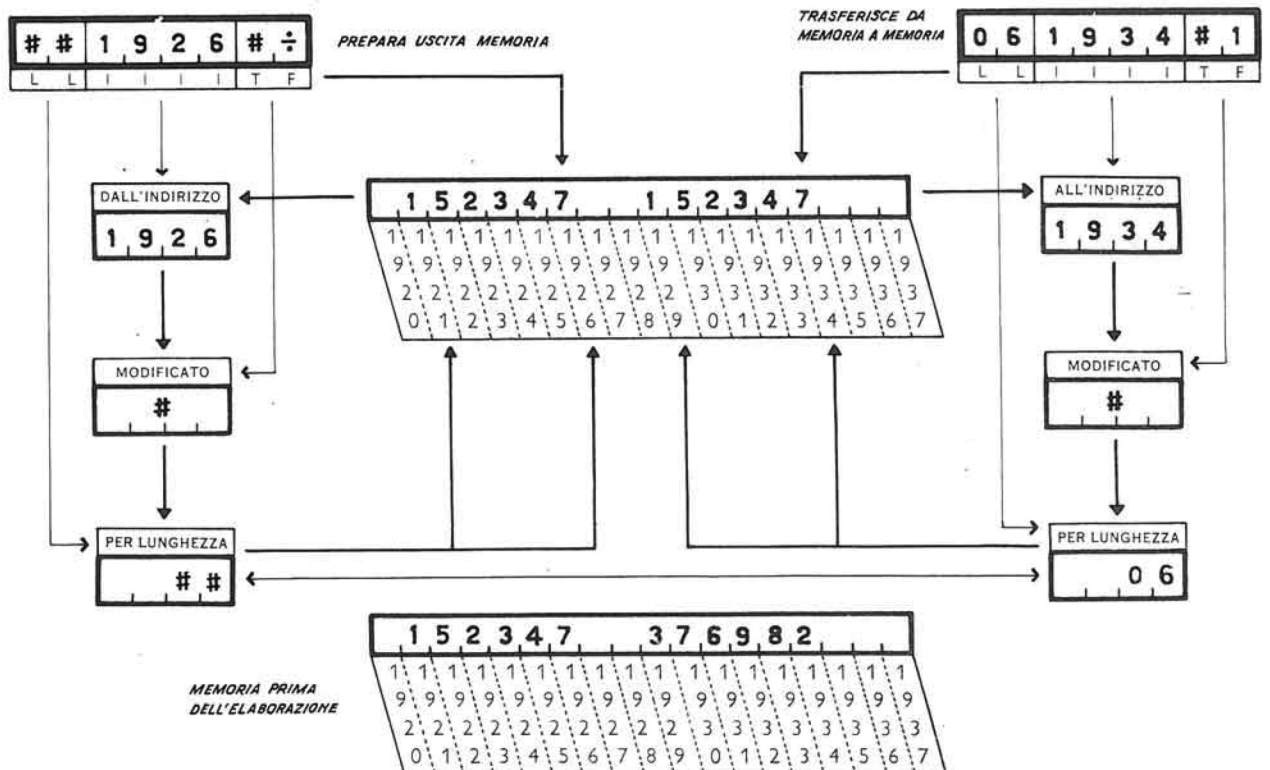
T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la lunghezza LL

1 : trasferisce la parola il cui indirizzo e' stato specificato nella PUM in altra zona di memoria a partire dall'indirizzo IIII specificato nella MEM per lunghezza LL e per indirizzi decrescenti.

a) Per questa istruzione valgono le osservazioni a, b, c, riguardanti la PUM.

b) La zona di memoria specificata dalla PUM rimane inalterata.

c) Contemporaneamente a questa istruzione non puo' essere eseguita alcuna istruzione che occupi almeno uno dei due canali.

d) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave: quest'ultimo deve essere contenuto nella zona di memoria con indirizzo specificato nella MEM.

Somma memoria a memoria			+MM
esterno ed interno	L L IIII T_m 3 (000111)		10 + 1

L L : lunghezza dei due operandi

I I I I : indirizzo di memoria del secondo operando

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

3 : somma il contenuto della memoria a partire dall'indirizzo IIII specificato nella PUM al contenuto della memoria a partire dall'indirizzo IIII specificato nella +MM per lunghezza LL e per indirizzi decrescenti.

a) Per questa istruzione valgono le osservazioni a, b, c, riguardanti la PUM.

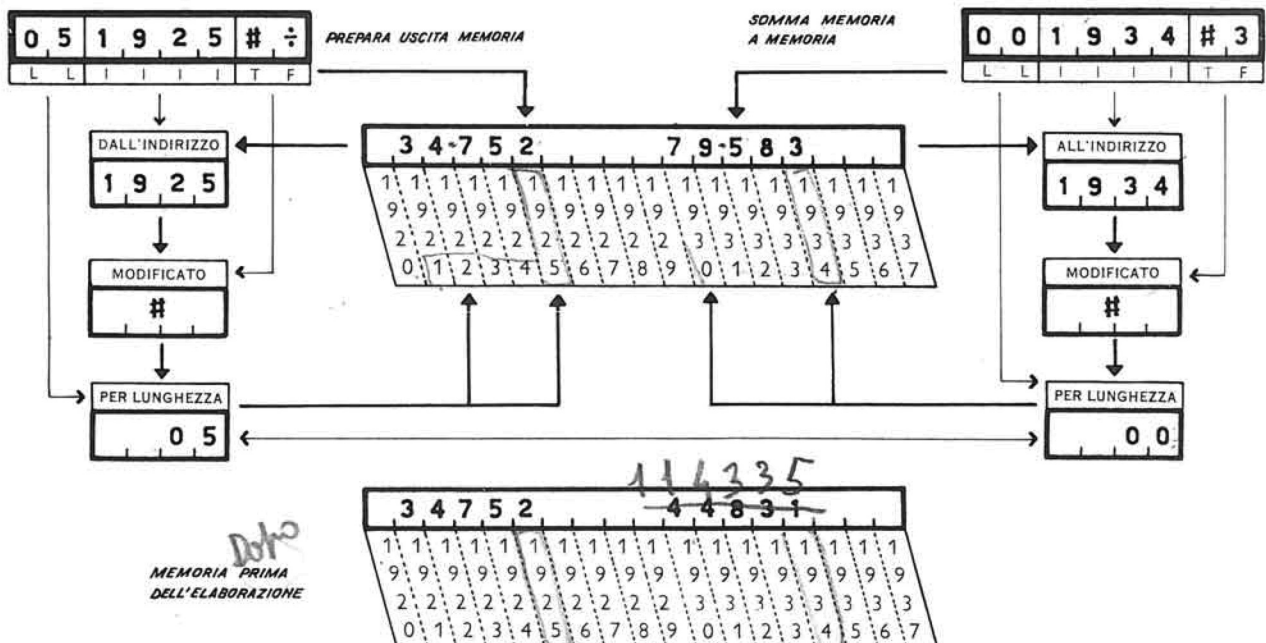
b) Il risultato si forma in memoria a partire dall'indirizzo IIII specificato nella presente istruzione.

c) Se il risultato e' di lunghezza superiore ad LL non vi e' riporto sulla posizione adiacente e vi e' indicazione di overflow.

d) La zona di memoria specificata nella PUM rimane inalterata.

e) L'operazione puo' avvenire solo per parole non segnate.

f) Contemporaneamente a questa istruzione non puo' essere eseguita alcuna istruzione che occupi almeno uno dei due canali.

g) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave: quest'ultimo deve essere contenuto nella zona di memoria con indirizzo specificato nella +MM.

Sottrae memoria da memoria			-MM
esterno ed interno	L L I I I I T_m 8 (001111)		10 ÷ 1

L L : lunghezza dei due operandi

I I I I : indirizzo di memoria del sottraendo

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

8 : sottrae il contenuto della memoria a partire dall'indirizzo IIII specificato nella presente istruzione al contenuto della memoria il cui indirizzo iniziale e' specificato nella PUM, per lunghezza LL e per indirizzi decrescenti.

a) Per questa istruzione valgono le osservazioni a, b, c, riguardanti la PUM.

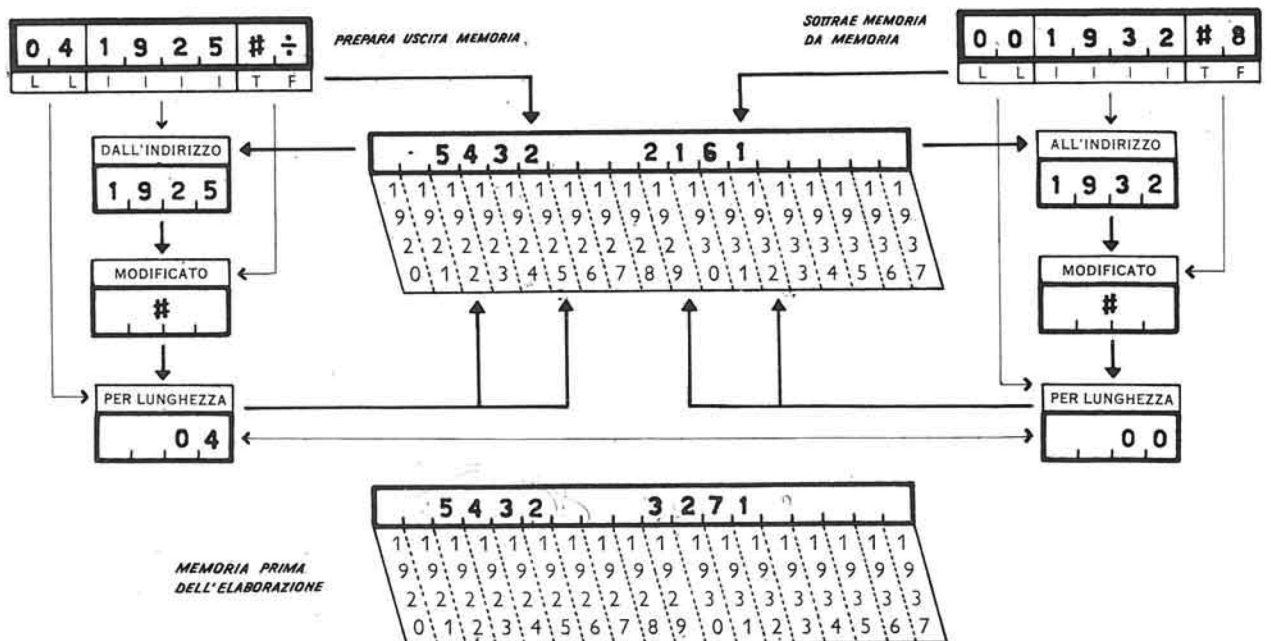
b) Il risultato si forma in memoria a partire dall'indirizzo IIII specificato nella presente istruzione.

c) La zona di memoria specificata nella PUM rimane inalterata.

d) L'operazione puo' avvenire solo per parole non segnate.

e) Qualora il minuendo sia minore del sottraendo il risultato appare in complemento e si ha indicazione di overflow.

f) Contemporaneamente a questa istruzione non puo' essere eseguita alcuna istruzione che occupi almeno uno dei due canali.

g) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave: quest'ultimo deve essere contenuto nella zona di memoria con indirizzo specificato nella -MM.

Confronta memoria con memoria		CMM
interno ed esterno	L L I I I I T_m / (001010)	10 + 1

L L : lunghezza dei due termini di confronto

I I I I : indirizzo del 1° termine di confronto

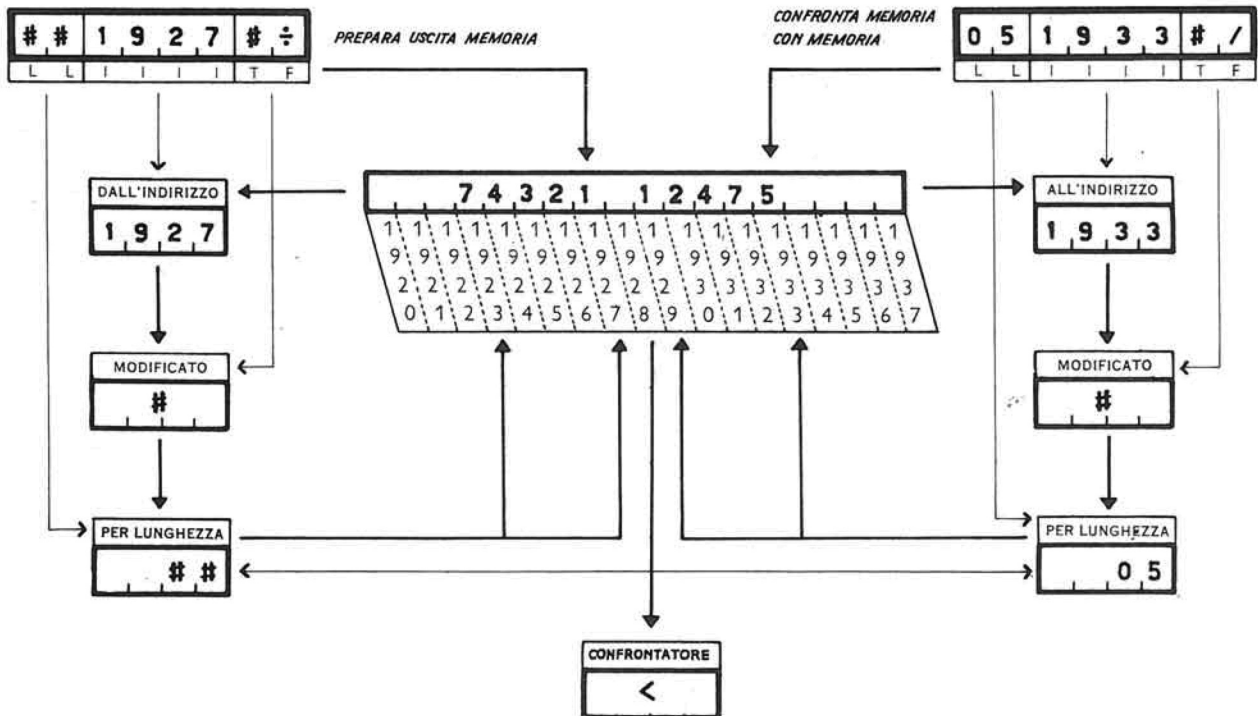
T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' di LL

/ : confronta il contenuto della memoria a partire dall'indirizzo IIII con il contenuto della memoria specificato nella PUM per lunghezza LL e per indirizzi decrescenti.

a) Per questa istruzione valgono le osservazioni a, b, c, riguardanti la PUM.

b) Il 1° termine di confronto e' quello indicato nella presente istruzione.

c) Le regole del confronto sono le stesse della CMA con l'avvertenza che il confronto può avvenire solo fra informazioni entrambe segnate o entrambe non segnate.

d) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave : quest'ultimo deve essere contenuto nella zona di memoria con indirizzo specificato nella CMM.

SCHEMA RIASSUNTIVO	Istruzioni Memoria Memoria

Trasferimento	:	PUM	MEM
Operazioni aritmetiche	:	PUM	+MM
		PUM	-MM
Confronto	:	PUM	CMM

Particolarita' della memoria centrale		CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE	CARATTERE DI FUNZIONE
Capacita'	: 20000 ÷ 40000 posizioni aumentabile a blocchi di 40000 fino a 160000 posizioni.	PUM	est.int.	10	L L I I I I T_m	÷
Memoria a nuclei magnetici	: ogni carattere e' materializzato dallo stato di 7 nuclei magnetici operati in parallelo.	MEM	" "	10 + 1	L L I I I I T_m	1
		+MM	" "	10 + 1	L L I I I I T_m	3
Tempo di accesso al carattere	: 10 microsecondi.	-MM	" "	10 + 1	L L I I I I T_m	8
Indirizzabilita'	al singolo carattere.	CMM	" "	10 + 1	L L I I I I T_m	/
Indirizzi	: sono espressi per mezzo di quattro cifre da 0000 a ''99 (vedi tabella del testo).					

7.3. Istruzioni memoria-registri

Queste istruzioni interessano la memoria principale ed i registri T : nel loro svolgimento impegnano la unita' aritmetica e logica ed il governo dell'elaboratore.

Sia durante la fase preparatoria che in quella esecutiva il canale interno risulta impegnato : a queste istruzioni potranno percio' sovrapporsi operazioni che impegnano il canale esterno ed il governo unita' a nastro.

Le istruzioni di questo gruppo sono normalmente utilizzate :

a) per trasferimenti, da memoria verso registri e viceversa, di parole segnate o non segnate e per lunghezza conosciuta o sconosciuta, non superiore a 10 caratteri; hanno questa funzione le istruzioni :

   MT, TM, ToM

b) per somme o sottrazioni aritmetiche su numeri di lunghezza variabile da 1 a 10 caratteri; eventuali segni algebrici + o -, vengono considerati caratteri qualsiasi; hanno questa funzione le istruzioni :

   + MT, - MT, + TM

c) per confronti tra numeri aritmetici o parole alfanumeriche di lunghezza non superiore a 10 caratteri; ha questa funzione l'istruzione :

   CMT

d) per determinare la lunghezza di una parola contenuta in un registro T ; ha questa funzione l'istruzione :

   FTM

Nota : La lunghezza della parola contenuta in un registro e' sempre espressa dalla istruzione FTM mediante due caratteri.
Si rende pertanto necessario un accorgimento di programmazione per posizionare tale lunghezza nell'unica posizione ad essa riservata nelle istruzioni da registro a memoria.
Generalmente tale lunghezza viene registrata in una zona di comodo della memoria principale per essere quindi trasferita per lunghezza uno (cifra di destra) nella posizione p8 dell'istruzione TM o ToM.

Trasferisce da memoria a registro T_o		MT
interno	L T_o I I I I T_m X (101001)	10 + 1

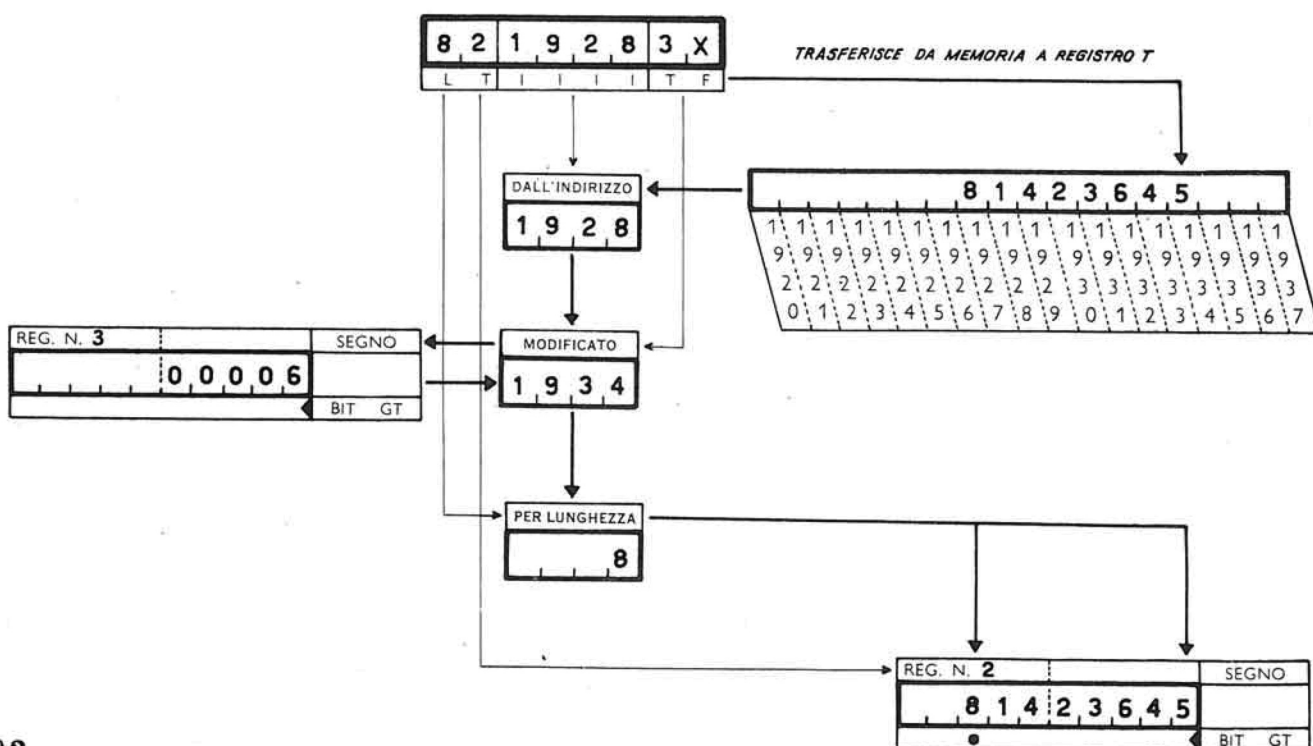
L	: lunghezza dell'operando da trasferirsi nel registro T_o
T_o	: registro in cui va trasferito l'operando
I I I I	: indirizzo di memoria dell'operando da trasferirsi
T_m	: il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro T_o
X	: trasferisce dalla memoria a partire dall'indirizzo IIII per lunghezza L e per indirizzi decrescenti, al registro T_o

a) I caratteri trasferiti si sostituiscono nel registro T_o a quelli precedentemente contenuti nelle posizioni interessate.

b) In corrispondenza dell'ultimo carattere trasferito viene posto un bit gT.

c) Se l'indirizzo IIII contiene il segno della parola, tale segno viene registrato nella prima posizione del registro T_o.

d) Il contenuto della memoria rimane inalterato.

e) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave; nel registro T_o, invece del carattere chiave, viene trasferito uno zero, ed il bit **gT** e' posizionato sotto quest'ultimo.

Trasferisce da registro T_o a memoria senza azzeramento		TM
interno	L T_o I I I I T_m Y (101011)	10 ÷1

L : lunghezza dell'operando da trasferirsi in memoria

T_o : registro da cui l'operando viene trasferito.

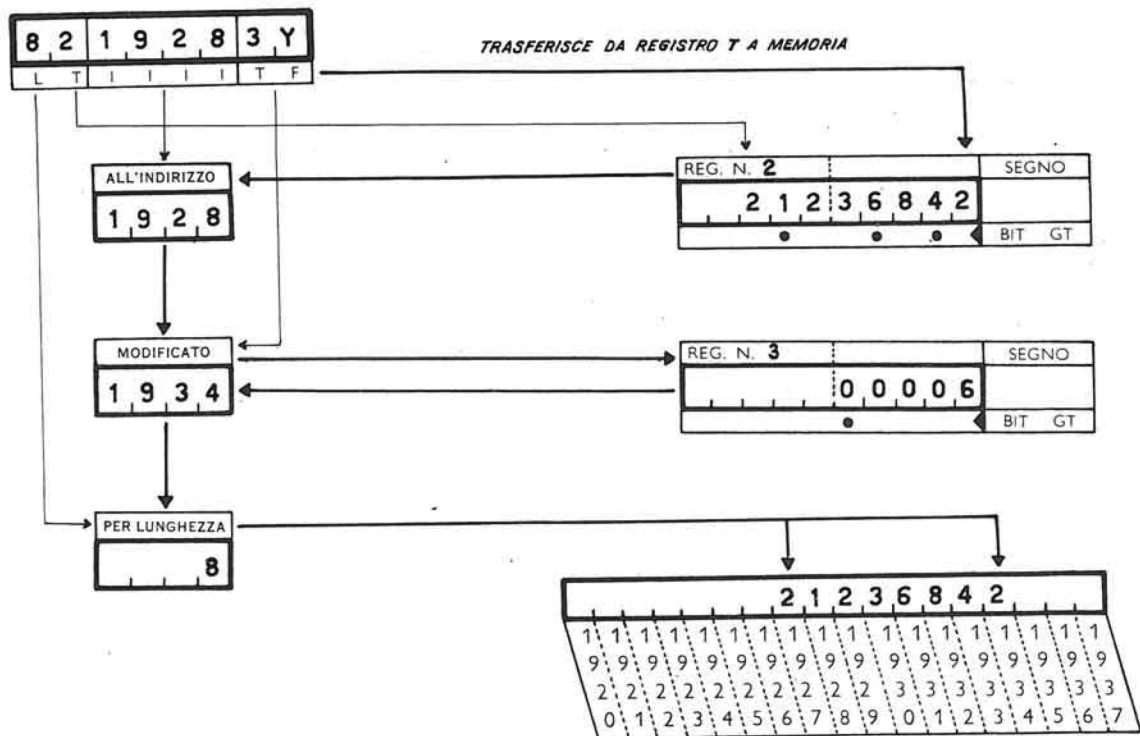
I I I I : indirizzo di memoria a cui viene trasferito l'operando contenuto nel registro T_o

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro T_o

Y : trasferisce il contenuto del registro T_o in memoria al l'indirizzo IIII per lunghezza L e per indirizzi decre scenti.

a) I caratteri trasferiti si sostituiscono in memoria a quelli precedentemente con tenuti nelle posizioni interessate.

b) Il contenuto del registro T_o rimane inalterato.

c) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave: quest'ultimo deve essere contenuto in memoria.

Trasferisce da registro T_O a memoria con azzeramento		ToM
interno	L T_O I I I I T_m T (100011)	10 ÷ 1

L : lunghezza dell'operando da trasferirsi in memoria

T_O : registro in cui e' registrato l'operando

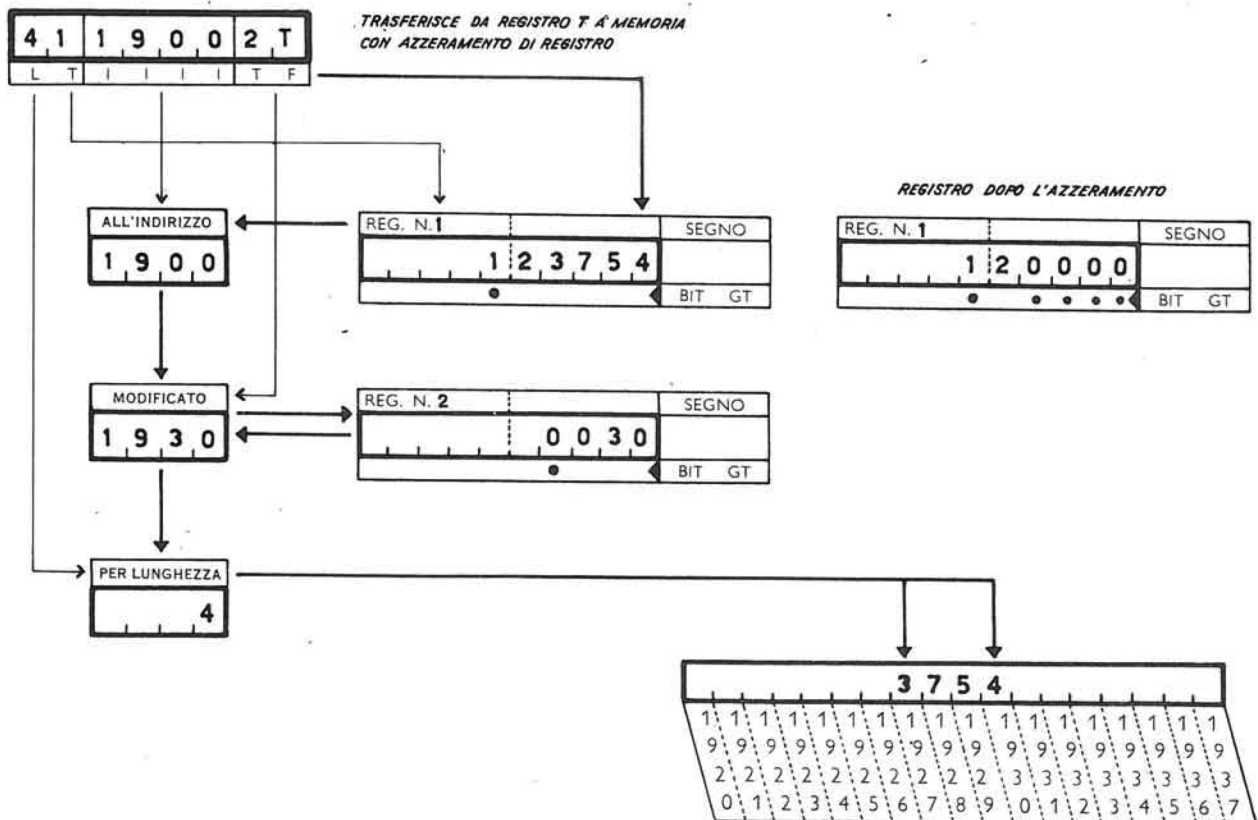
I I I I : indirizzo di memoria a cui viene trasferito l'operando

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro T_O

T : trasferisce dal registro T_O alla memoria dall'indirizzo IIII per lunghezza L e per indirizzi decrescenti

a) A trasferimento avvenuto le posizioni del registro T_O interessate sono azzerate.

b) In ognuna di queste posizioni e' posto un bit gT.

c) Per questa istruzione valgono le regole relative ai casi di fine su carattere speciale : quest'ultimo deve essere contenuto in memoria.

Somma memoria a registro T_0, risultato in registro T_0		+ MT
interno	L T_0 I I I I T_m U (100111)	10 ÷ 1

L : lunghezza dell'operando registrato in memoria

T_0 : registro che contiene il secondo operando

I I I I : indirizzo dell'operando registrato in memoria

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro T_0

U : somma il contenuto della memoria a partire dall' indirizzo IIII per lunghezza L e per indirizzi decrescenti al contenuto del registro T_0 compreso tra la posizione iniziale e il bit gT

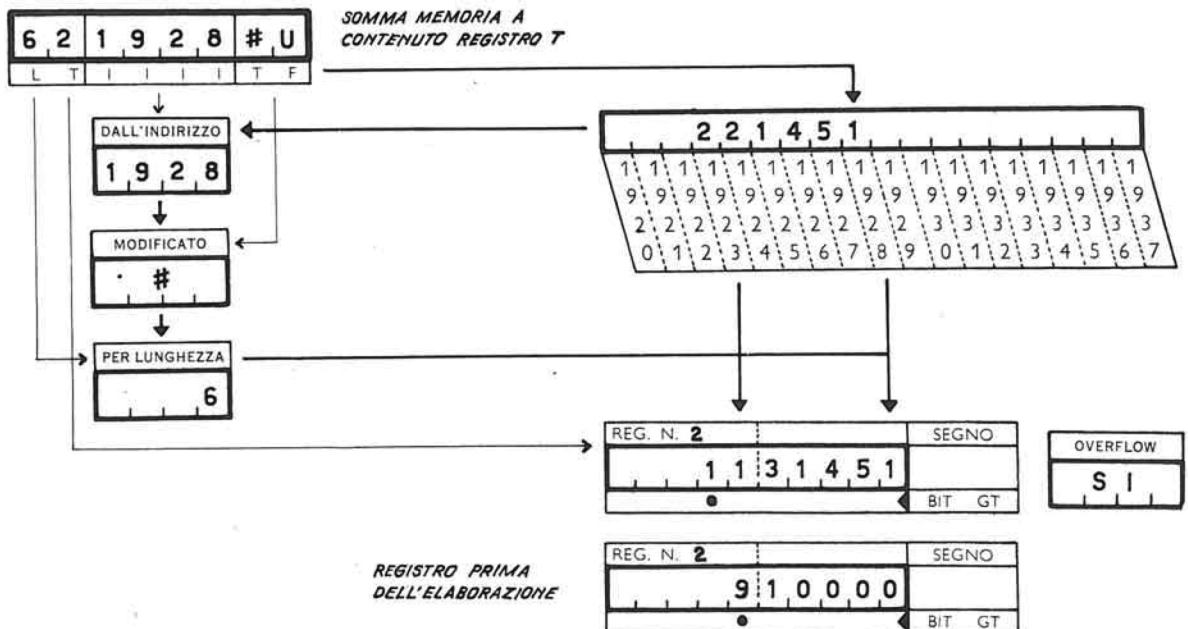
a) Il bit gt viene posto in corrispondenza del carattere piu' significativo del risultato.

b) Se il risultato in caso di riporto e' di lunghezza maggiore dei due operandi si ha indicazione di overflow.

c) Un eventuale riporto dalla decima posizione invade il registro adiacente e si ha indicazione di overflow.

d) Se uno degli operandi e' espresso mediante caratteri alfanumerici si puo' operare correttamente su di esso fino a che non esista la necessita' di un riporto oltre la parte numerica.

e) Non e' possibile mediante la presente istruzione ottenere risultati alfanumerici utilizzabili quali indirizzi di memoria.

f) Per questa istruzione vale solo il caso di fine su carattere non numerico : il carattere Z o Y deve essere posto in p8.

Somma registro T_o a memoria, risult. in memoria			+ TM
interno	L T_o I I I I T_m ⊠ (101110)		10 + 1

L	:	lunghezza dell'operando registrato in memoria
T_o	:	registro che contiene il secondo operando
I I I I	:	indirizzo dell'operando registrato in memoria
T_m	:	il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro T_o
⊠	:	somma il contenuto del registro T_o compreso fra la posizione iniziale e il bit gT al contenuto della memoria a partire dall'indirizzo IIII per lunghezza L e per indirizzi decrescenti.

a) Se L e' maggiore del numero delle posizioni del registro T_o comprese tra la posizione iniziale ed il bit gT, le cifre oltre il bit gT non vengono operate.

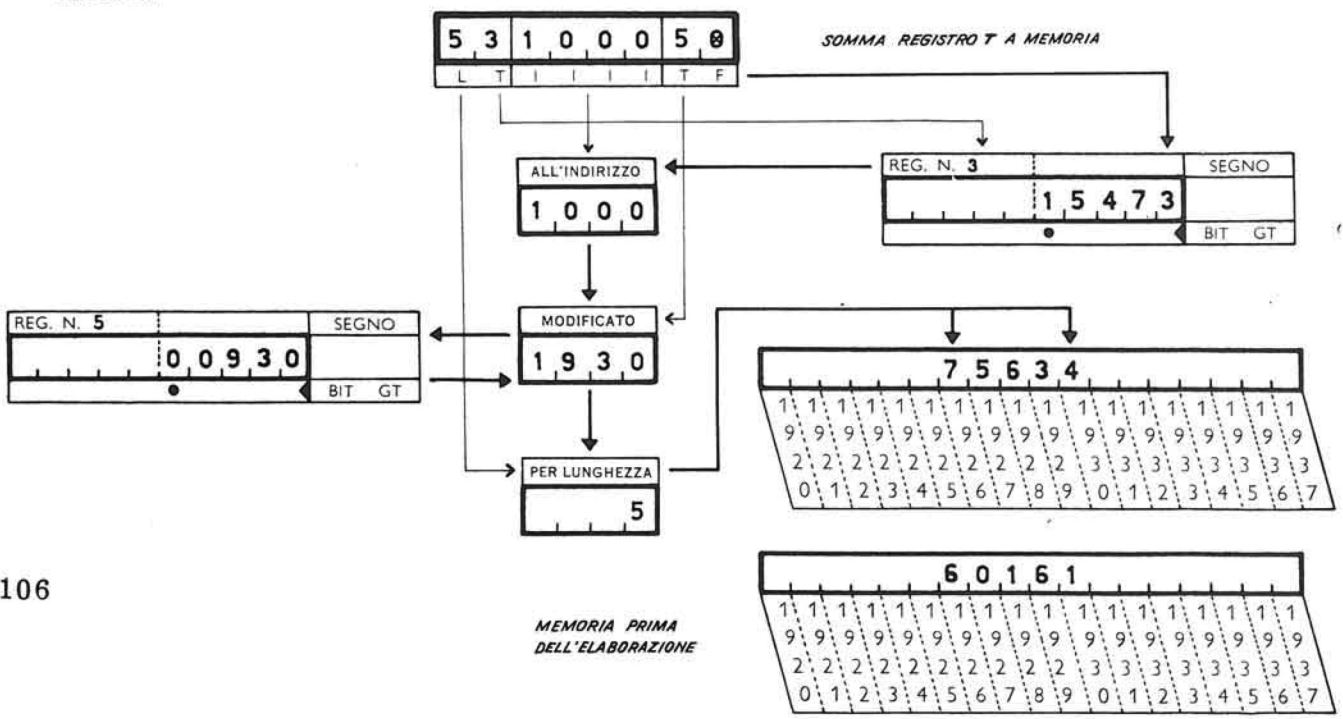
b) Se L e' minore del numero delle posizioni del registro T_o comprese tra la posizione iniziale ed il bit gT, le cifre oltre la lunghezza L non vengono operate e si ha indicazione di overflow.

c) Il risultato si forma in memoria a partire dall'indirizzo IIII.

d) Il registro T_o rimane invariato.

e) Il risultato non deve superare la lunghezza L.

f) Non si ha l'eventuale riporto oltre la lunghezza L ma solo indicazione di overflow.

g) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave.

Sottrae memoria a registro T_0, risultato in registro T_0		-MT
interno	L T_0 I I I I T_m Z (101111)	10 + 1

L : lunghezza dell'operando registrato in memoria

T_0 : registro che contiene il secondo operando

I I I I : indirizzo di memoria a partire dal quale e' registrato il sottraendo

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro T_0

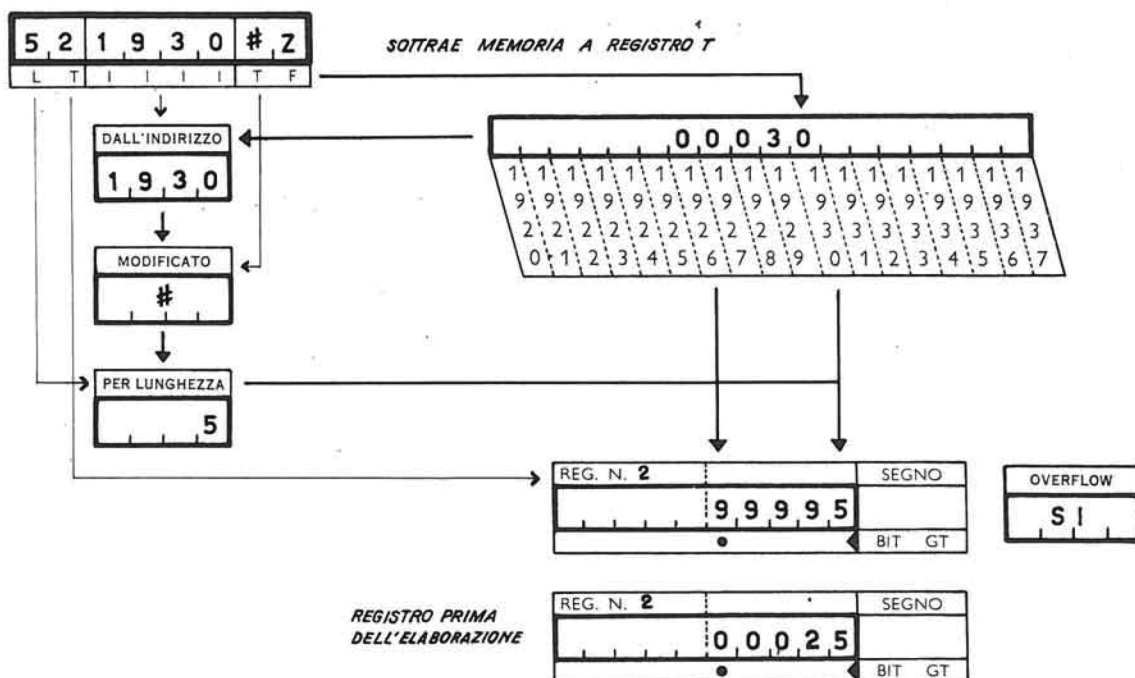
Z : sottrae il contenuto della memoria a partire dall'indirizzo IIII per lunghezza L, al contenuto del registro T_0 compreso fra la posizione iniziale e il bit gT

a) Il bit gT viene posto in corrispondenza del carattere piu' significativo del risultato.

b) Se il minuendo e' minore del sottraendo il risultato e' in complemento e si ha indicazione di overflow.

c) Non e' possibile mediante la presente istruzione ottenere risultati alfanumerici utilizzabili quali indirizzi di memoria.

d) Se uno degli operandi e' espresso mediante caratteri alfanumerici si puo' operare correttamente su di esso fino a che non esista la necessita' di un riporto oltre la parte numerica.

e) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave, ponendo in p8 il carattere Z o Y; e' evidente che non e' possibile il caso di fine su di un carattere ben determinato.

Confronta memoria con registro T_0		CMT
interno	L T_0 I I I I T_m W (101000)	10 ÷ 1

L : lunghezza del termine di confronto contenuto in memoria

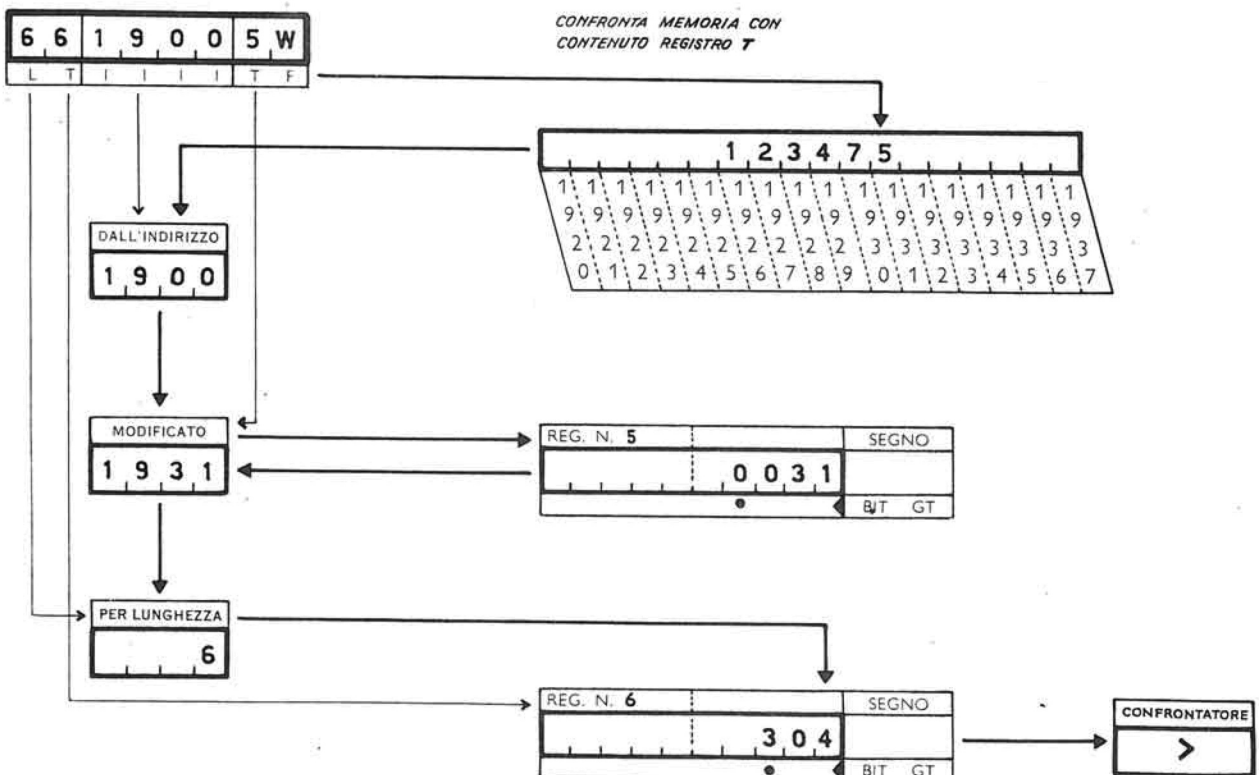
T_0 : registro che contiene il secondo termine di confronto

I I I I : indirizzo di memoria del termine registrato in memoria

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro T_0

W : confronta il contenuto della memoria a partire dall'indirizzo IIII per lunghezza L, con il contenuto del registro compreso tra la posizione iniziale e il bit gT

a) Per questa istruzione vale cio' che e' stato detto per la CMA ad eccezione dei segni "piu' o meno", che non vengono mai considerati come segni algebrici ma come caratteri.

b) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave : quest'ultimo deve essere contenuto in memoria.

Fine registro T_0 in memoria		FTM
interno	0 T_0 I I I I T_m θ (100010)	12 + 1

0 : carattere che determina il genere di lunghezza ricercata

T_0 : registro che contiene l'operando di cui si vuole determinare la lunghezza

I I I I : indirizzo al quale va trasferita la lunghezza ricercata

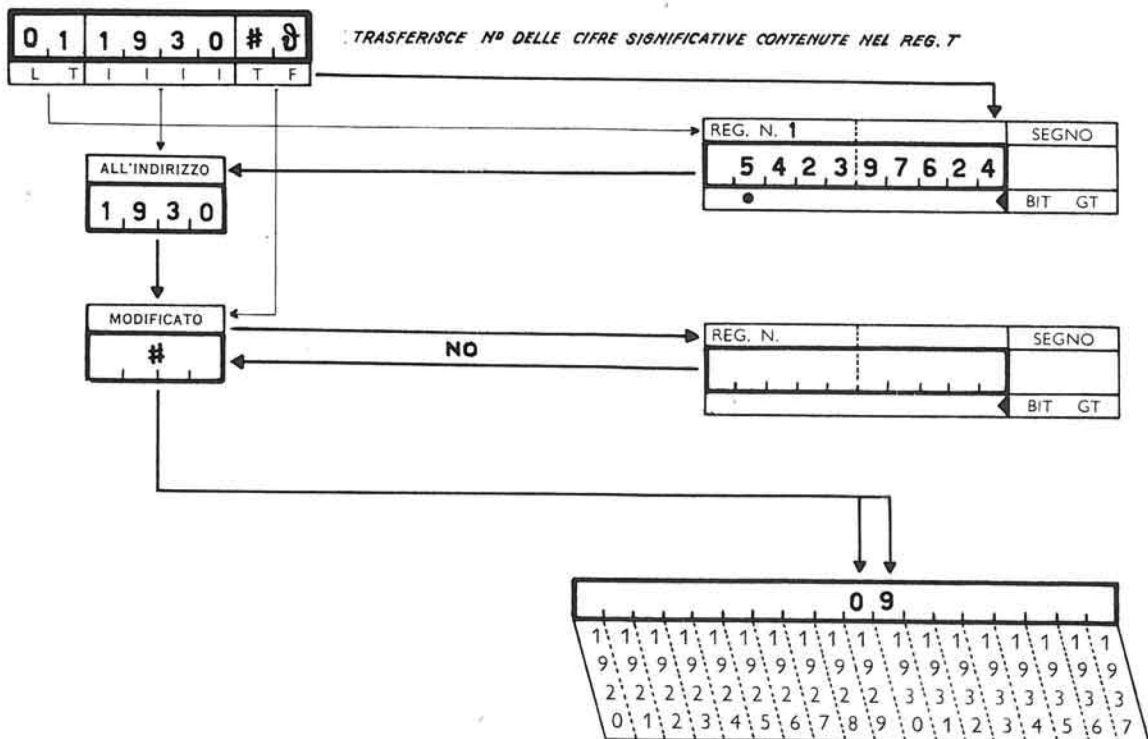
T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro T_0

θ : porta all'indirizzo IIII il numero di cifre contenute nel registro T_0 dalla posizione iniziale al bit gT.

a) Se in p8 viene registrato il carattere 0 (zero), viene trasferito all'indirizzo IIII il numero delle cifre significative comprese tra la posizione iniziale del registro T_0 e il bit gT.

b) Se in p8 viene registrato il carattere . (punto), viene trasferito all'indirizzo IIII il numero delle cifre significative e non significative comprese tra la posizione iniziale del registro T_0 e il bit gT.

c) La lunghezza richiesta viene registrata sempre mediante 2 cifre.

d) Se richiedendo il numero di cifre significative (caso "a") non ne risultasse alcuna, vengono trasferiti all'indirizzo IIII e IIII-1, due zeri (00).

SCHEMA RIASSUNTIVO	Istruzioni Memoria Registri

Trasferimento	:	MT	TM	ToM
Operazioni aritmetiche	:	+MT	+TM	-MT
Confronto	:	CMT		
Determinazione lunghezza	:	FTM		

Particolarita' dei Registri

Posizioni : 200
n^{o} dei registri : 40
Indirizzabilita' : di 5 in 5 posizioni indicando il nome del registro.

Tabella nome registri : 0123456789ØABCDEFGHI
ƐJKLMNOPQR STUVWXYZ

Capacita' del singolo registro : 10 posizioni.

Modifica Automatica Istruzioni :

a) posizioni dell'istruzione interessata p7 p6 p5 p4 p3 ;

b) posizioni del registro interessato 5 4 3 2 1 .

La modifica avviene sommando i caratteri del registro a quelli della istruzione nelle posizioni suindicate senza riporto tra p6 e p7.
La somma nella 4^{a} posizione si effettua secondo una aritmetica in base venti.
Il riporto di p7 si somma al carattere p8 dell'istruzione mentre l'eventuale riporto di p8 si perde.

Bit gT : analogo al bit gA dell'accumulatore. Per il suo funzionamento vedere istruzione per istruzione.

CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE	CARATTERE DI FUNZIONE
MT	interno	10 + 1	L T_o I I I I T_m	X
TM	"	10 + 1	L T_o I I I I T_m	Y
ToM	"	10 + 1	L T_o I I I I T_m	T
+MT	"	10 + 1	L T_o I I I I T_m	U
+TM	"	10 + 1	L T_o I I I I T_m	Ø
-MT	"	10 + 1	L T_o I I I I T_m	Z
CMT	"	10 + 1	L T_o I I I I T_m	W
FTM	"	12 + 1	0 T_o I I I I T_m	Θ
FTM	"	12 + 1	. T_o I I I I T_m	Θ

7.4. Istruzioni costanti-registri

Queste istruzioni interessano parole costituite da caratteri o valori fissi (costanti); nel loro svolgimento impegnano l'unita' aritmetica e logica ed il governo dell'elaboratore.

Sia durante la fase preparatoria che in quella esecutiva il canale interno risulta impegnato; a queste istruzioni potranno percio' sovrapporsi operazioni che impegnano il canale esterno ed il governo unita' a nastro.

Nella loro posizione p8 e' necessario registrare il carattere ÷ oppure ≠ ; gli eventuali segni, piu' o meno, vengono considerati caratteri qualsiasi.

Esse non possono subire modifica automatica ne' permettono l'introduzione nei registri T dei caratteri ≠ e ÷ ; sono normalmente utilizzate :

a) per trasferimenti nei registri T di costanti di lunghezza non superiore a 5 caratteri; ha questa funzione l'istruzione :

   CT

b) per somme o sottrazioni aritmetiche di lunghezza non superiore a 5 caratteri; istruzioni di questo tipo sono :

   + CT, - CT, CTT

c) per confronti tra numeri aritmetici o parole alfanumeriche di lunghezza variabile da 1 a 5 caratteri; istruzione di questo tipo e' la :

   CCT

<table>
<tr><td colspan="2">Trasferisce una costante nel registro T_0</td><td>CT</td></tr>
<tr><td>interno</td><td>≠ C CCCC T_0 9 (001100)</td><td>10</td></tr>
</table>

≠ : posizione non utilizzata

C C C C C : posizioni riservate alla costante da registrarsi nel registro T_0

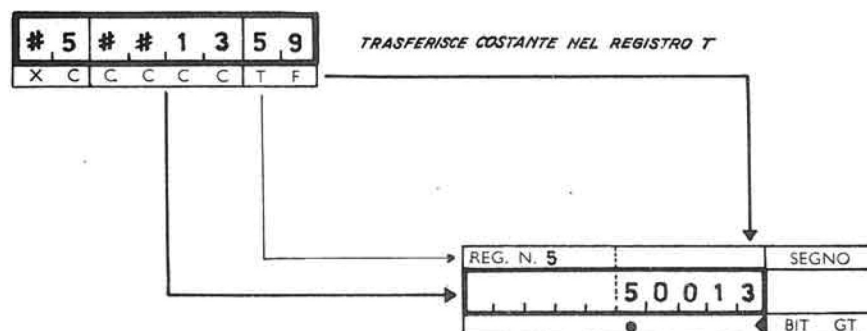
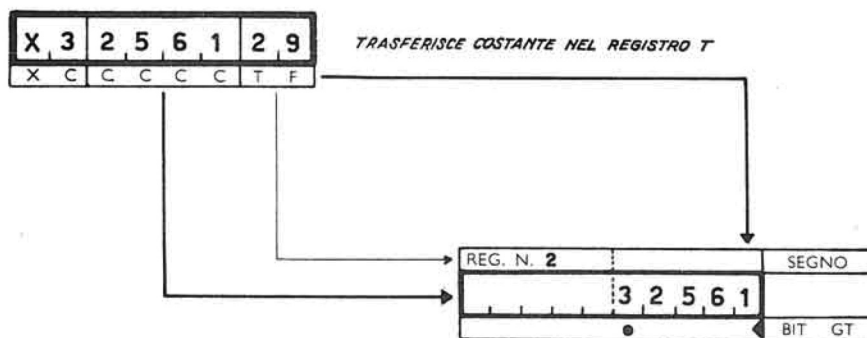
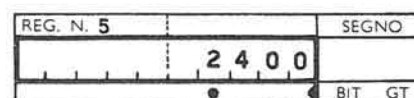
T_0 : registro in cui viene registrata la costante

9 : trasferisce nel registro T_0 la costante CCCCC registrata nell'istruzione.

a) Il bit gT viene posto in corrispondenza della 5ª posizione del registro T_0.

b) Se nelle posizioni p7 e p8 e' registrato il carattere ÷ il trasferimento avviene solo per la costante CCCC ed il bit gT e' posto in 4ª posizione.

REGISTRO PRIMA DELL'ELABORAZIONE

REG. N. 5 — SEGNO — 2 4 0 0 — BIT GT

tipo A	Somma una costante a registro T_0	+ CT
interno	÷ ÷ C C C C T_0 + (000101)	10

÷ ÷	:	determinano il tipo d'istruzione + CT
C C C C	:	costante da sommarsi al contenuto del registro T_0
T_0	:	registro su cui si opera
+	:	somma la costante CCCC al contenuto del registro T_0.

a) Il bit gT si trova nel registro T_0 prima dell'istruzione in 5ª posizione.

b) La somma viene effettuata tra i primi 5 caratteri del registro T_0 e la costante CCCC .

c) Il risultato e' espresso per mezzo di 4 caratteri di cui il 4° puo' essere alfabetico o speciale.

d) Il contenuto della 5ª Posizione del registro T_0 rimane inalterato.

e) Non vi e' indicazione di overflow.

f) Il bit gT viene posto in 4ª posizione.

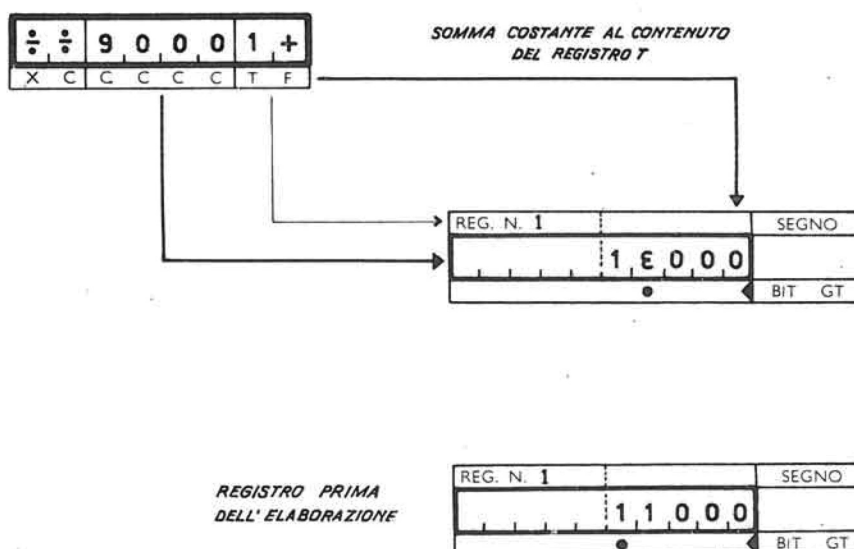
g) Ponendo 0000 in CCCC e' possibile trasformare un indirizzo espresso con 5 caratteri numerici inizialmente contenuto nel registro T_0 in un indirizzo di 4 caratteri con le migliaia espresse nel modo corretto.

h) E' possibile percorrere tutti i successivi indirizzi di memoria.

REGISTRO PRIMA DELL' ELABORAZIONE

REG. N. 1 | SEGNO

1 1 0 0 0

BIT GT

tipo B	Somma una costante a registro T_0	÷ CT

interno	÷ ÷ C C C C T_0 + (000101)	10

÷ ÷ : determinano il tipo di istruzione ÷ CT

C C C C : costante da sommarsi al contenuto del registro T_0

T_0 : registro su cui si opera

÷ : somma la costante CCCC al contenuto del registro T_0.

a) Il bit gT si trova nel registro T_0 prima dell'istruzione in 4ª posizione.

b) La somma viene effettuata sui primi quattro caratteri del registro T_0.

c) Il 4° carattere puo' essere, o risultare dopo la somma, alfabetico o speciale.

d) Non vi e' mai riporto sulla quinta posizione del registro T_0, ne' indicazione di overflow.

e) Il bit gT rimane in 4ª posizione.

f) E' possibile percorrere tutti i successivi indirizzi di memoria.

tipo C	Somma una costante a registro T_0	+ CT

interno	# # C C C C T_0 + (000101)	10

: determinano il tipo di istruzione + CT

C C C C : costante da sommarsi al contenuto del registro T_0

T_0 : registro su cui si opera

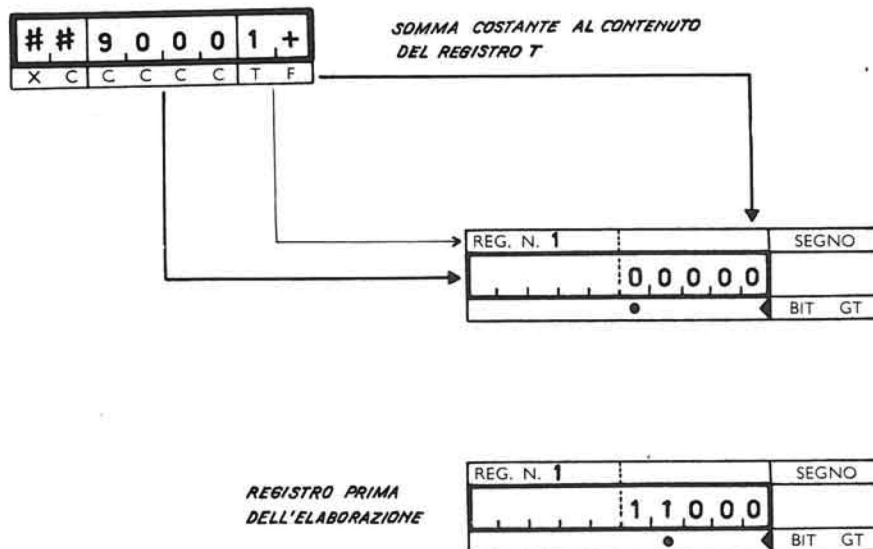
+ : somma la costante CCCC al contenuto del registro T_0.

a) Il bit gT si trova nel registro T_0 prima dell'istruzione in 4^a posizione.

b) La somma viene effettuata sui primi quattro caratteri del registro T_0.

c) Non vi e' mai riporto sulla quinta posizione del registro T_0, ne' indicazione di overflow.

d) Il bit gT viene posto in 5^a posizione, ed in essa viene generato il carattere 0 (zero).

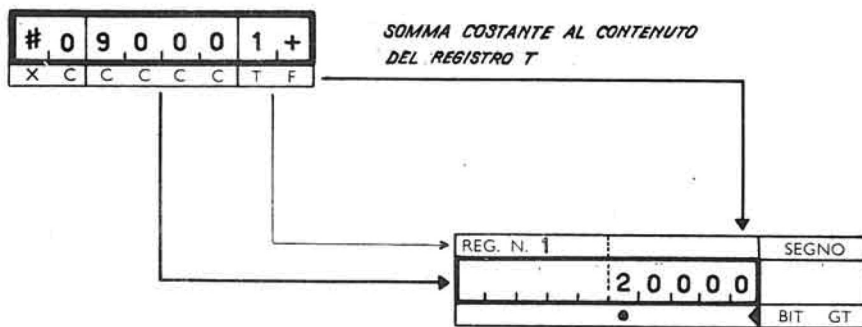
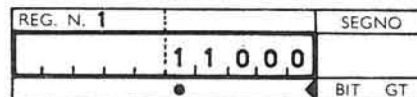
e) Non e' possibile percorrere tutti i successivi indirizzi di memoria non esistendo interazione tra parte numerica e parte alfabetica.

tipo D	Somma una costante a registro T_0	+ CT

interno	# [C CCCC] T_0 + (000101)	10

#	: posizione non utilizzata
C C C C C	: costante da sommarsi al contenuto del registro T_0
T_0	: registro su cui si opera
+	: somma la costante CCCCC al contenuto del registro T_0.

a) La somma viene effettuata tra la costante CCCCC e il contenuto del registro T_0 compreso tra la posizione iniziale e il bit gT

b) Nel caso vi fosse riporto oltre la quinta posizione del registro si ha indicazione di overflow.

c) Il bit gT dopo l'operazione viene sempre posto in 5ª posizione.

Sottrae una costante a registro T_0			- CT
interno	# C C C C C T_0 4 (000100)		10

#	: posizione non utilizzata
C C C C C	: costante da sottrarsi al contenuto del registro T_0
T_0	: registro su cui si opera
4	: sottrae la costante CCCCC al contenuto del registro T_0.

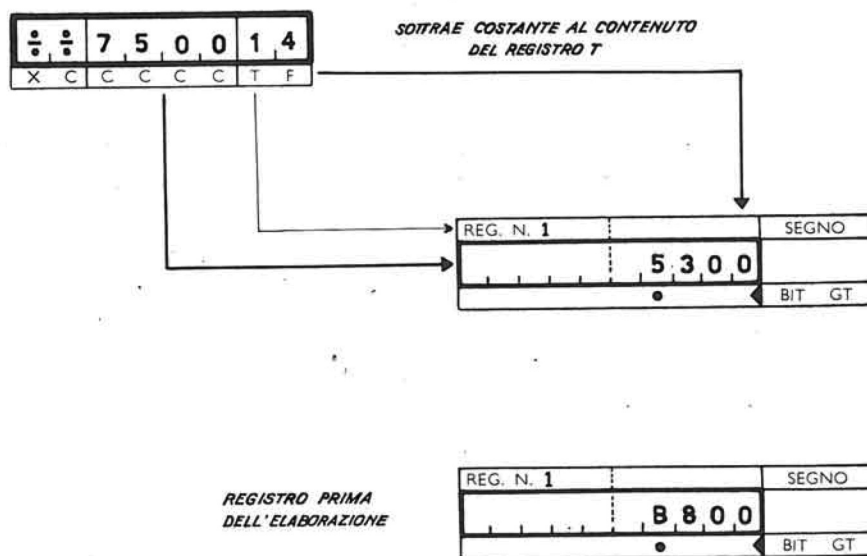
a) L'istruzione - CT puo' essere codificata anche nei seguenti modi :

C C C C T_o 4

÷ ÷ C C C C T_o 4

b) Il tipo con il carattere ÷ in p7 e p8 vale solo nel caso che, prima dell'istruzione, il bit gT sia posto in 4ª posizione.

c) Per l'istruzione - CT valgono le stesse osservazioni della + CT.

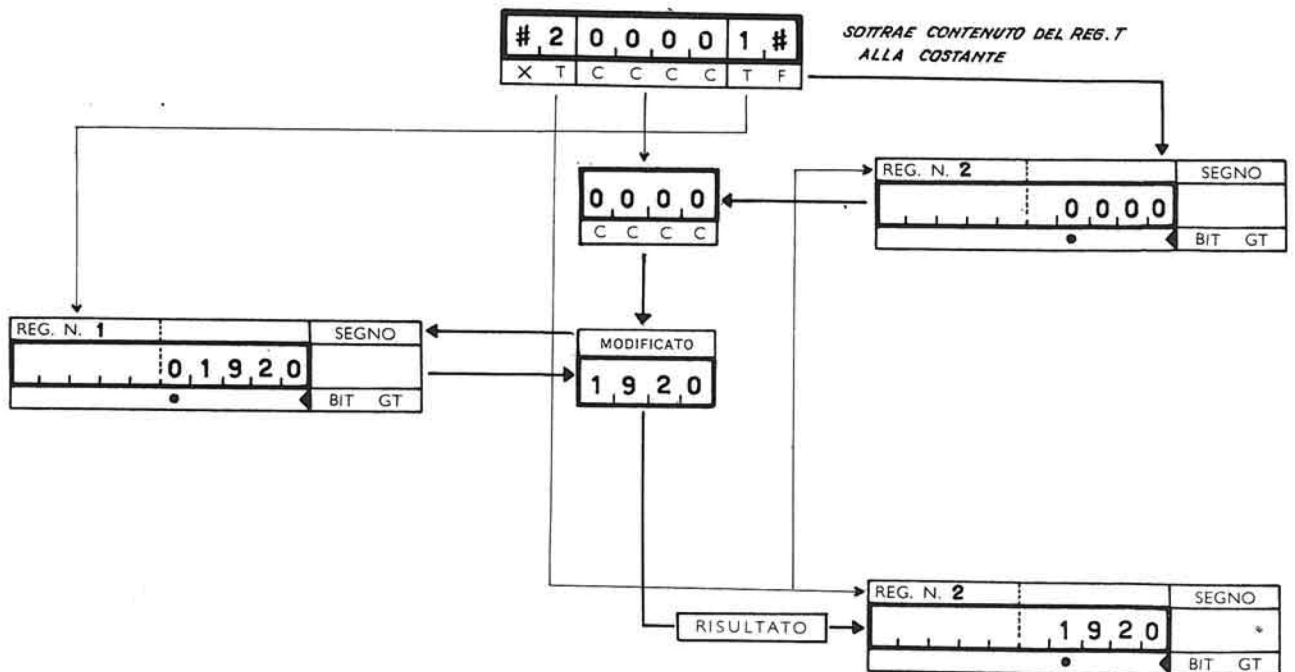
Esempio della nota b)

Sottrae registro T_o ad una costante, risultato in T_o		CTT
interno	# T_o C C C C T_m # (000110)	14

#	:	posizione non utilizzata
T_o	:	registro che contiene il sottraendo
C C C C	:	minuendo
T_m	:	puo' modificare sia CCCC che il nome del registro T_o
#	:	sottrae il contenuto delle prime quattro posizioni del registro T_o alla costante CCCC modificabile da T_m.

a) Il risultato della sottrazione e' nel registro T_o.

b) Il contenuto della quinta posizione di T_o rimane inalterato ed il bit gT viene in ogni caso posto in 4^a posizione.

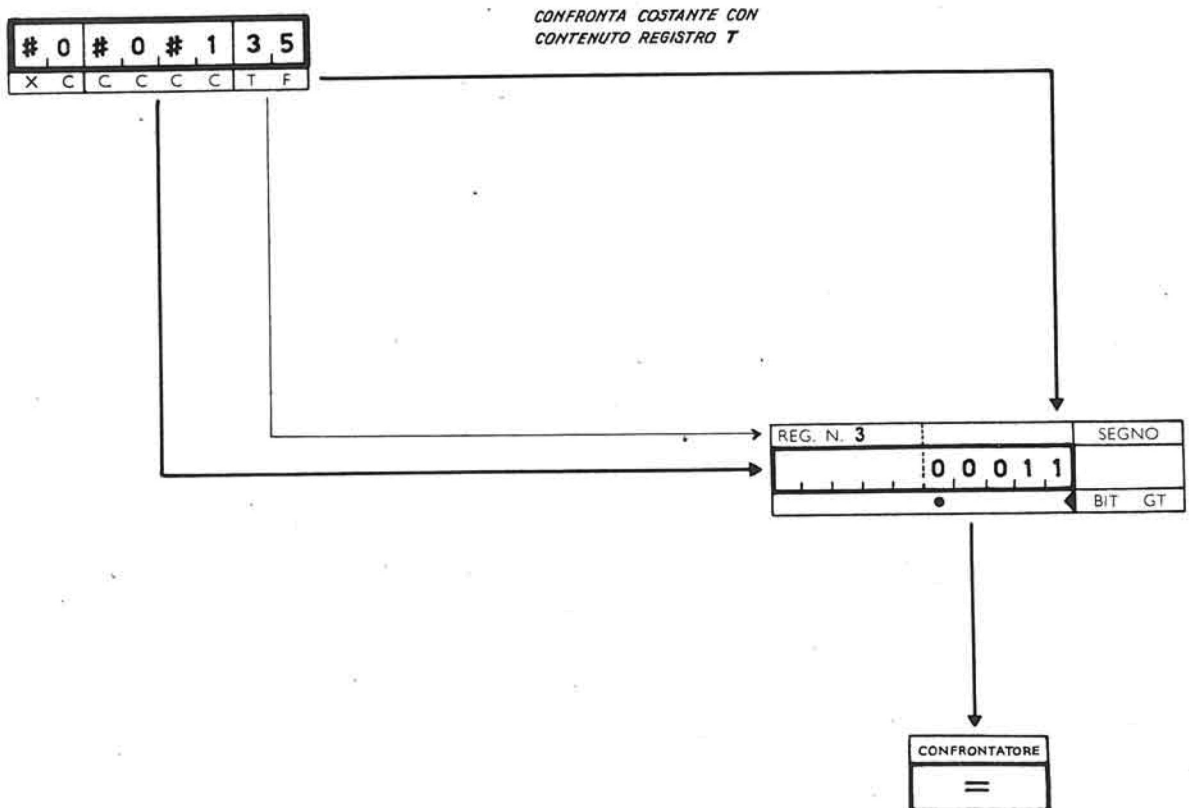
c) Per una corretta esecuzione della presente istruzione e' necessario che in CCCC e nel registro T_o non compaiano caratteri alfabetici o speciali.

d) 1°) Se T_m + CCCC > 9999 il riporto oltre la 4^a cifra non viene considerato;
   2°) Se $T_o > T_m$ + CCCC in T_o si ha, a seguito dell'operazione, un risultato in complemento.

Confronta una costante con registro T_0		CCT
interno	# C C C C C T_0 5 (001000)	10

#	:	posizione non utilizzata
C C C C C	:	1° termine di confronto
T_0	:	nel registro T_0 e' registrato il secondo termine di confronto
5	:	confronta la costante CCCCC al contenuto del registro T_0 dall'inizio e fino al bit gT per un massimo di 5 posizioni.

a) Le regole del confronto sono le stesse della istruzione CMA ad eccezione per quello che riguarda i segni + e - che non vengono mai considerati come segni algebrici, ma come caratteri.

b) Se si desidera che il confronto non tenga conto di tutte le cinque posizioni della costante basta registrare il carattere # nelle posizioni che non interessano.

SCHEMA RIASSUNTIVO	Istruzioni Costanti Registri

Trasferimento : CT

Operazioni aritmetiche : +CT -CT CTT

Confronto : CCT

Particolarita' delle istruzioni su costanti	CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE	CARATTERE DI FUNZIONE
Il registro T_0 specificato nella posizione p2 non opera come modificatore; fa eccezione la CTT che ha in p2 il registro modificatore e in p7 il registro operando.	CT	interno	10	# C C C C C T_0	9
	"	"	"	÷ ÷ C C C C T_0	9
Le istruzioni di questo tipo non sono automaticamente modificabili.	+CT	"	10	÷ ÷ C C C C T_0	+
	"	"	"	÷ ÷ C C C C T_0	+
In corrispondenza di p8 va sempre messo il carattere # o ÷.	"	"	"	# # C C C C T_0	+
	"	"	"	# C C C C C T_0	+
Il tempo necessario all'esecuzione di un'istruzione di questo tipo e' 10 periodi di cifra, esclusa la CTT alla quale ne sono necessari 14.	-CT	"	10	# C C C C C T_0	4
	"	"	"	÷ ÷ C C C C T_0	4
	"	"	"	# # C C C C T_0	4
Bit gT : analogo al bit gA dell'accumulatore. Per il suo funzionamento vedere istruzione per istruzione.	CTT	"	14	# T_0 C C C C T_m	#
	CCT	"	10	# C C C C C T_0	5

7.5. Istruzioni per la moltiplicazione

Queste istruzioni interessano la memoria principale, i registri T e l'accumulatore; nel loro svolgimento impegnano l'unita' aritmetica e logica ed il governo dell'elaboratore.

Sia durante la fase preparatoria che in quella esecutiva il canale interno risulta impegnato; a questo gruppo di istruzioni potranno percio' sovrapporsi operazioni che impegnano il canale esterno ed il governo unita' a nastro.

Esse sono normalmente utilizzate :

a) per il trasferimento nella memoria dei registri T, a partire dalla posizione "00" e per indirizzi crescenti, del moltiplicatore, per lunghezza conosciuta o sconosciuta variabile da 1 a 100 caratteri; l'eventuale segno algebrico va a posizionarsi in un apposito registro del segno; ha questa funzione l'istruzione :

Y

b) per sommare o sottrarre, automaticamente, il risultato della moltiplicazione al contenuto dell'accumulatore; questa funzione e' propria delle istruzioni :

+ X, - X

Trasferisce il moltiplicatore nei registri T	Y

interno	L L I I I I T_m 6 (001001)	10 + 1

L L : lunghezza del moltiplicatore contenuto in memoria

I I I I : indirizzo in memoria del moltiplicatore

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

6 : trasferisce la parola contenuta in memoria all'indirizzo IIII per lunghezza LL nei registri T a partire da T0.

a) La cifra meno significativa della parola occupa la posizione iniziale 0 del registro T0 e successivamente in ordine progressivo vengono occupate le altre posizioni.

b) Possono essere eventualmente invase tutte le posizioni del gruppo di registri compresi tra T0 e TI.

c) La lunghezza della parola puo' essere al massimo eguale a 100 (LL = 00).

d) Se la parola e' segnata il segno non viene registrato nella prima posizione di T0, ma in un apposito registro.

e) Se la parola non e' segnata e non supera lunghezza 10 l'istruzione Y puo' essere sostituita con una qualsiasi delle istruzioni relative ai registri T.

f) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave; nel caso di trasferimento lungo piu' di 90, non deve essere usato il registro T9.

Nei due esempi che seguono, il moltiplicatore viene trasferito nei registri T tramite le istruzioni Y ed MT.

Come si puo' osservare e' possibile ottenere il trasferimento del segno nell'apposito registro solo mediante l'istruzione Y.

TRASFERISCE DA MEMORIA MOLTIPLICATORE
1 2 1 9 3 2 # 6
L L I I I I T F
DALL'INDIRIZZO
1 9 3 2
MODIFICATO
#
PER LUNGHEZZA
1 2
1 2 4 5 3 6 1 2 7 8 9 +
REG. N. 1
REG. N. 0
SEGNO
1 2 4 5 3 6 1 2 7 8 9
+
BIT GT
ESATTO

TRASFERISCE DA MEMORIA A REGISTRO T0
3 0 1 9 3 2 # X
L T I I I I T F
DALL'INDIRIZZO
1 9 3 2
MODIFICATO
#
PER LUNGHEZZA
3
8 9 +
REG. N. 0
SEGNO
8 9 +
BIT GT
ERRATO

Moltiplicazione additiva	+ X

interno	L L I I I I T_m = (010101)	10 + n° cifre Mol. rex(3+1)

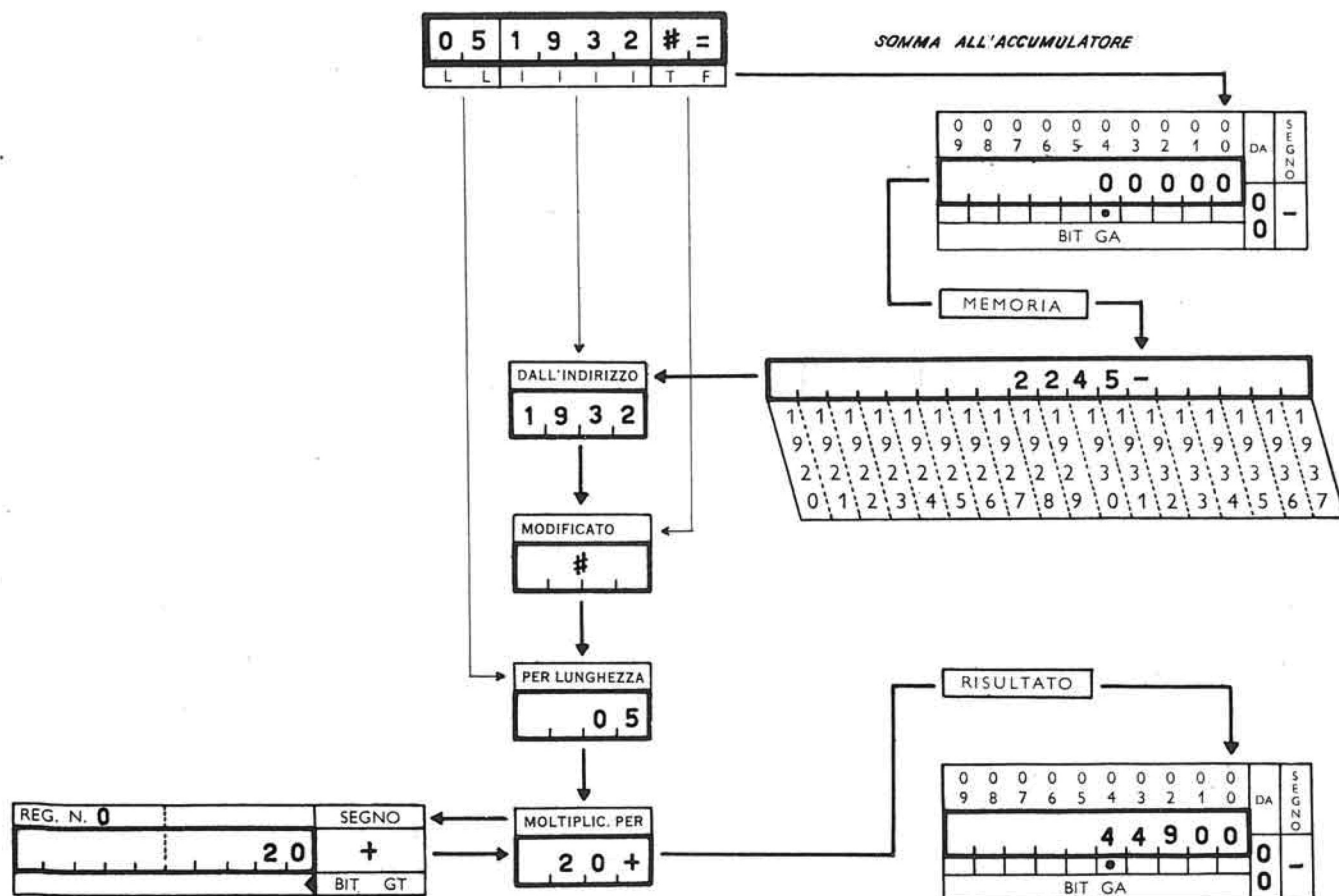
L L	:	lunghezza del moltiplicando
I I I I	:	indirizzoin memoria del moltiplicando
T_m	:	il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL
=	:	somma al contenuto dell'accumulatore il risultato della moltiplicazione.

a) La moltiplicazione viene effettuata tra il contenuto della memoria a partire dal l'indirizzo IIII e per lunghezza LL, e il contenuto dei registri dalla posizione 0 (T0) fino al primo bit gT.

b) Il contenuto della memoria e dei registri rimane inalterato.

c) Affinche' il risultato sia segnato e' sufficiente che lo sia il contenuto della memoria o quello dell'accumulatore.

d) Non e' sufficiente che sia segnato solo il moltiplicatore.

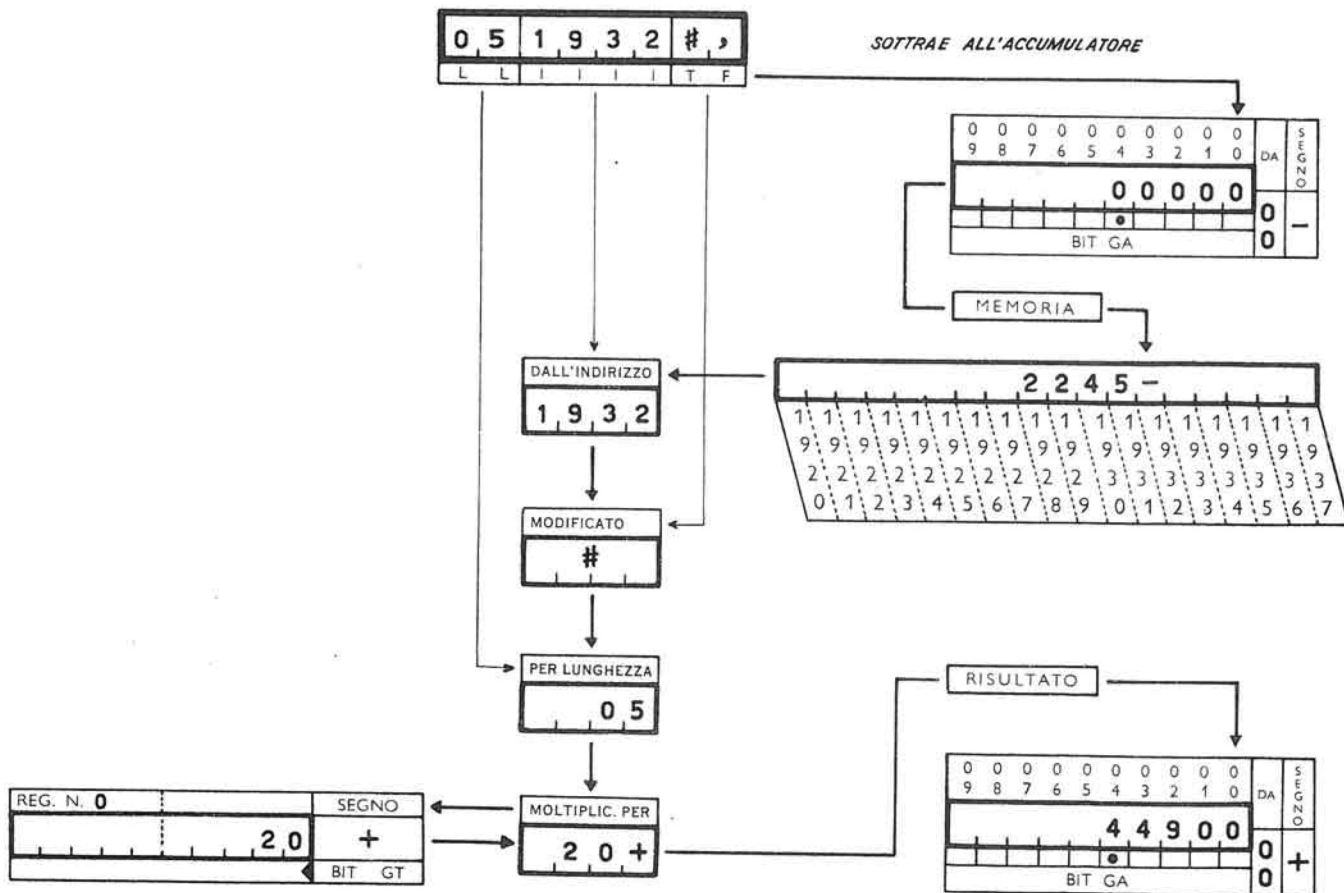
e) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave.

Moltiplicazione sottrattiva			- X
interno	L L IIII T_m , ,	(011101)	$10+n^o$ cifre Mol.re x (3+1)

L L : lunghezza del moltiplicando

I I I I : indirizzo in memoria del moltiplicando

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che le cifre delle unita' della lunghezza LL

, : sottrae al contenuto dell'accumulatore il risultato della moltiplicazione.

a) Per l'istruzione - X valgono le osservazioni fatte per la +X.

SCHEMA RIASSUNTIVO — Istruzioni di Moltiplicazione

Trasferimento : Y

Moltiplicazione : +X -X

Particolarita' della moltiplicazione

Due fasi : Registrazione del moltiplicatore : istruzione Y .

Esecuzione vera e propria : istruzione +X , -X .

La moltiplicazione viene eseguita tra due numeri segnati o non segnati dei quali uno (il moltiplicatore) contenuto nei registri T a partire da TO, e l'altro (il moltiplicando) in una posizione qualsiasi di memoria.

Il trasferimento del moltiplicatore in TO deve necessariamente farsi con la Y solo quando e' un numero segnato o di lunghezza maggiore di 10.

Il risultato viene sommato o sottratto al contenuto dell'accumulatore.

Il tempo richiesto per la fase esecutiva delle istruzioni +X e -X e' :

$$L\ (3 + l)$$

dove L e' il numero delle cifre del moltiplicatore ed l il numero delle cifre del moltiplicando.

CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE	CARATTERE DI FUNZIONE
Y	interno	10 + l	L L I I I I T_m	6
+X	"	10 +(3+l) L	L L I I I I T_m	=
-X	"	10 +(3+l) L	L L I I I I T_m	,

7.6. Istruzioni per la ricerca in memoria

Le istruzioni di ricerca interessano la memoria principale; nel loro svolgimento impegnano l'unita' aritmetica e logica ed il governo dell'elaboratore.

Sia nella fase preparatoria che in quella esecutiva il canale interno risulta impegnato. A questo gruppo di istruzioni potranno percio' sovrapporsi istruzioni interessanti il canale esterno ed il governo unita' a nastro.

Queste istruzioni consentono di trovare l'indirizzo della posizione di memoria che contiene un carattere determinato oppure un carattere appartenente ad una classe fissa, e di ricordarlo in un registro T.

La ricerca viene effettuata da una posizione iniziale di memoria, procedendo sia per indirizzi crescenti (RIa) sia per indirizzi decrescenti (RIi).

L'indirizzo della posizione di memoria contenente il carattere ricercato viene registrato nelle prime quattro posizioni del registro T9.

La ricerca avviene confrontando il contenuto della memoria con il carattere posto in R.
Di questo carattere possono essere interessati dal confronto tutti o parte dei bit.

Il numero e la posizione dei bit da confrontare vengono allora determinati dai bit 1 di un secondo carattere posto in C.

Se ne deduce che se il carattere posto in C e' Q (111111) tutti i bit di R vengono confrontati con i bit dei caratteri esistenti in memoria, e che un solo carattere puo' dare eguaglianza nel confronto: il carattere identico ad R.

Se invece il carattere posto in C e' diverso da Q, per l'esclusione di almeno un bit dal confronto, si crea un gruppo di caratteri pari alla potenza di 2 elevato ad esponente uguale al numero dei bit 0 di C

	f e d c b a
Esempio: In p8 sia posto il carattere F	(0 1 1 0 0 1)
e in p7 sia posto il carattere ε	(1 1 0 0 0 0)

La macchina ricerchera' un carattere che abbia i bit "e" ed "f" rispettivamente uguali a "1" e "0". Il numero dei caratteri aventi queste caratteristiche e' $2^4 = 16$.

Ricerca avanti			RIa
interno	R C I I I I T_m 2 (000011)		15 + 1

R : carattere o classe di carattere ricercati

C : carattere che condiziona i bit di R interessati dalla ricerca

I I I I : indirizzo da cui si inizia la ricerca

2 : ricerca a partire dall' indirizzo IIII e procedendo per indirizzi crescenti l' indirizzo della prima posizione di memoria nella quale sia registrato un carattere appartenente alla classe individuata da R e C.

a) L'indirizzo della posizione di memoria contenente il carattere ricercato viene registrato nelle prime quattro posizioni del registro T9.
b) Se il carattere ricercato si trova all'indirizzo IIII indicato nell' istruzione viene registrato in T9 l'indirizzo IIII.
c) Se il carattere posto in C e' Q (111111), tutti i bit intervengono nella ricerca e viene quindi cercato solo il carattere posto in R.
d) Se il carattere posto in C e' diverso da Q, si ricerca una classe di caratteri che abbiano in comune i bit corrispondenti indicati da C.
e) Il carattere "1" indicato nella casella riservata alla durata dell'istruzione si riferisce al numero di caratteri confrontati prima di raggiungere quello desiderato.

Ricerca indietro			RIi
interno	R C I I I I T_m 7 (001011)		15 + 1

R : carattere o classe di caratteri ricercati

C : carattere che condiziona i bit di R interessati dalla ricerca

I I I I : indirizzo da cui si inizia la ricerca

7 : ricerca a partire dall' indirizzo IIII e procedendo per indirizzi decrescenti l' indirizzo della prima posizione di memoria nella quale sia registrato un carattere appartenente alla classe individuata da R e C.

a) Valgono le osservazioni fatte a proposito della RIa.

SCHEMA RIASSUNTIVO	Istruzioni di Ricerca

Istruzioni : RIa RIi

Particolarita' istruzioni di ricerca	CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE	CARATTERE DI FUNZIONE
Funzione : consentono di determinare l'indirizzo della posizione di memoria che contiene un dato carattere o un carattere appartenente ad una classe fissata; l'indirizzo del carattere e' trasferito nel registro T9.	RIa	interno	15 + l	R C I I I I T_m	2
La ricerca viene effettuata da una posizione iniziale per indirizzi crescenti o decrescenti.					
La ricerca viene eseguita confrontando il contenuto della memoria con il carattere posto in p8 dell'istruzione; vengono interessati dal confronto soltanto i bit corrispondenti a quelli che nel carattere posto in p7 sono uguali ad 1.	RIi	"	15 + l	R C I I I I T_m	7
Tempo : 15 + l periodi di cifra dove l e' il numero di caratteri confrontati prima di raggiungere quello desiderato.					

7.7. Le istruzioni per le operazioni logiche

Il calcolatore e' capace di eseguire operazioni logiche, bit per bit, secondo l'algebra di Boole, fra un operando proveniente da memoria e un altro proveniente da un registro T.

L'operazione puo' eseguirsi sia tra i bit diretti, sia tra i bit "negati" che si ottengono prendendo i bit componenti un carattere e scambiando gli 1 con 0 e gli 0 con 1.

Cosi' ad esempio del carattere F i bit diretti sono 011001, i bit negati sono 100110.

La lunghezza degli operandi nelle operazioni logiche non puo' essere superiore a 10.

Anche queste istruzioni possono operare su dati di lunghezza sconosciuta con fine determinabile da parola chiave; sono pero' previsti solo i due casi di fine su segno o carattere alfabetico.

Il carattere indicativo del tipo di fine, deve essere specificato nella posizione p8 dell'istruzione .

E' evidente che se l'istruzione comporta l'uso del T9 non e' possibile la fine con trasferimento dell'indirizzo della parola chiave. Occorre pure ricordare che queste istruzioni hanno significato se la lunghezza e' ≤ 10 ; quindi il segno o carattere alfabetico non deve trovarsi in memoria oltre la decima posizione toccata.

Si tenga pero' conto che se la parola del registro e' piu' corta di quella di memoria si ha fine prima della comparsa del carattere chiave, e in T9 si trasferisce eventualmente l'indirizzo dell'ultimo carattere di memoria operato.

Se invece la fine avviene per il carattere speciale, l'ultimo periodo operativo e' precedente alla comparsa di detto carattere. Nel periodo del carattere chiave si ha scrittura in T di un zero con pallinaccio (g).

Moltiplicazione logica diretta		XLD
interno	L T_o I I I I T_m ! (100101)	10 + 1

L : lunghezza degli operandi

T_o : registro in cui si trova il secondo operando

I I I I : indirizzo di memoria del primo operando

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro operando T_o

! : effettua la moltiplicazione logica, bit per bit tra i bit diretti dei caratteri costituenti la parola contenuta in memoria di lunghezza L ed i bit diretti dei caratteri contenuti nelle prime L posizioni del registro T_o.

a) Il risultato si forma in T_o.

b) Il contenuto della memoria rimane inalterato.

c) Le regole secondo le quali si esegue la moltiplicazione logica diretta sono date dalla seguente tabella :

2° \ 1°	0	1
0	0	0
1	0	1

d) Per questa istruzione valgono le regole relative ai casi di fine carattere chiave.

Moltiplicazione logica negata			XLN
interno	L T_o I I I I T_m & (101101)		10 + 1

L : lunghezza degli operandi

T_o : registro in cui si trova il secondo operando

I I I I : indirizzo di memoria del primo operando

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro operando T_o

& : effettua la moltiplicazione logica bit per bit, tra i bit negati dei caratteri costituenti la parola contenuta in memoria e di lunghezza L e i bit negati dei caratteri contenuti nelle prime L posizioni di T_o.

a) Il risultato si forma in T_o.

b) Il contenuto della memoria rimane inalterato.

c) Le regole secondo le quali si esegue la moltiplicazione logica negata sono date dalla seguente tabella :

1° \ 2°	0	1
0	1	0
1	0	0

d) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave.

Somma logica diretta			+LD
interno	L T_o I I I I T_m V (100100)		10 + 1

L : lunghezza degli operandi

T_o : registro in cui si trova il secondo operando

I I I I : indirizzo di memoria del primo operando

T_m : puo' modificare sia l'indirizzo IIII che il nome del registro operando T_o

V : effettua la somma logica, bit per bit, tra i bit diretti dei caratteri costituenti la parola contenuta in memoria e di lunghezza L ed i bit diretti dei caratteri contenuti nelle prime L posizioni del registro T_o.

a) Il risultato si forma in T_o.

b) Il contenuto della memoria rimane inalterato.

c) Le regole secondo le quali si esegue la somma logica diretta sono date dalla seguente tabella :

1° \ 2°	0	1
0	0	1
1	1	1

d) Per questa istruzione valgono le regole relative ai casi di fine su carattere chiave.

SCHEMA RIASSUNTIVO	Operazioni logiche

Somma : + LD

Moltiplicazione : XLD XLN

Particolarita' delle operazioni logiche	CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE	CARATTERE DI FUNZIONE
Operandi : uno in memoria e uno in un registro T.					
Operazioni : secondo l'algebra di Boole bit per bit; le operazioni possono avvenire sia tra bit diretti che tra bit negati.	XLD	interno	10 + 1	L T_o I I I I T_m	!
	XLN	"	10 + 1	L T_o I I I I T_m	&
Bit diretti : bit operati con il valore attuale nella configurazione del carattere.	+LD	"	10 + 1	L T_o I I I I T_m	V
Bit negati : ottenuti dai bit diretti invertendo il valore degli "1" con "0" e degli "0" con "1".					
Lunghezza degli operandi : non puo' essere superiore a 10.					
Risultato : nel registro T_o gia' contenente uno degli operandi.					

7.8. Istruzioni relative al tavolo di comando

Queste istruzioni possono interessare con la memoria principale o il fotolettore o la telescrivente; durante il loro svolgimento impegnano l'unita' aritmetica e logica ed il governo dell'elaboratore.

Appartengono a questo gruppo :

le istruzioni CM (da telescrivente o fotolettore a memoria), MS (da memoria a telescrivente).

La CM, sia durante la fase preparatoria che in quella esecutiva, tiene impegnato il canale interno; a questa istruzione potranno percio' sovrapporsi operazioni che impegnano il canale esterno ed il governo unita' a nastro.

La MS impegna il canale interno per tutto il tempo richiesto dalla fase preparatoria; durante la fase esecutiva invece quest'ultimo viene impegnato dall'istruzione ogni settimo di secondo per il tempo necessario al trasferimento di un carattere.

Le istruzioni di questo gruppo sono normalmente utilizzate :

a) per introdurre nella memoria principale, mediante telescrivente, parole di lunghezza variabile da 1 a 100 caratteri, oppure mediante fotolettore, parole di lunghezza qualsiasi;

b) per estrarre dalla memoria principale, mediante telescrivente, parole di lunghezza variabile da 1 a 100 caratteri.

Da tavolo di comando o da fotolettore a memoria		CM
interno	L L IIII T_m % (011010)	10

L L : lunghezza dell'informazione da trasferirsi in memoria

I I I I : indirizzo a partire dal quale i caratteri registrati vengono trasferiti in memoria

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' di LL

% : legge da tastiera o da fotolettore verso memoria.

a) La scelta fra tastiera e fotolettore e' fatta mediante la chiave TAST - FL posta sul tavolo di comando.

b) La lettura da tastiera avviene per LL caratteri via via che i rispettivi tasti vengono premuti.

c) La lettura da fotolettore avviene :

1) per un numero di caratteri pari a quello indicato in LL ;

2) ponendo ## al posto di LL la lettura avviene per tutti i caratteri registrati su banda perforata fino al primo carattere α (che viene anch'esso registrato in memoria).

d) Sono previsti anche per questa istruzione i due casi di fine su carattere chiave (non si ha pero' trasferimento in T9 dell'indirizzo del carattere chiave).

e) Il senso di registrazione in memoria e' per indirizzi crescenti.

f) Il tempo indicato nella casella della durata dell'istruzione si riferisce solo alla fase preparatoria; la fase esecutiva avviene alla velocita' di 800 caratteri al secondo.

Trasferisce da memoria a telescrivente		MS
interno	L L IIII T_m Ø (010000)	10

L L : lunghezza dell'informazione da stampare

I I I I : indirizzo di memoria in cui e' registrato il 1° carattere da trasferire

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

Ø : scrive con la telescrivente LL caratteri a partire da quello che si trova all'indirizzo IIII e per indirizzi crescenti.

a) Nello svolgimento del programma in cui e' posta l'istruzione si intercalano periodi di telescrivente, in ognuno dei quali si comanda la stampa di un carattere letto in memoria.

b) Alla fine del trasferimento si esegue automaticamente un ritorno carrello e interlinea; se LL e' maggiore di 72, si torna automaticamente a capo.

c) Se durante l'esecuzione di questa istruzione viene eseguito lo STOP, questa non viene interrotta ma prosegue fino alla fine.

d) Il tempo indicato nella casella della durata dell'istruzione, si riferisce alla fase preparatoria; la fase esecutiva avviene alla velocita' di 7 caratteri al secondo.

SCHEMA RIASSUNTIVO	Istruzioni relative al tavolo di comando

Trasferimento : CM MS

Particolarita' istruzioni relative al tavolo di comando	CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE	CARATTERE DI FUNZIONE
Funzione : consentono di introdurre o di estrarre dei dati in o da memoria principale.					
L'introduzione : puo' avvenire tramite la telescrivente per un massimo di 100 caratteri, oppure per mezzo del fotolettore fino al primo carattere α perforato su nastro.	CM	interno	10	L L I I I I T_m	%
L'estrazione : avviene solo mediante telescrivente per un numero di caratteri non superiore a 100.	MS	"	10	L L I I I I T_m	Ø
Tempo : 10 periodi di cifra per la fase preparatoria; la fase esecutiva viene eseguita alla velocita' di 800 caratteri al secondo per la CM, usando il fotolettore, e di 7 caratteri al secondo per la MS.					

7.9. Istruzioni di salto

La successione delle istruzioni di programma puo' dover essere alterata al verificarsi di certe eventualita'. A questo scopo servono le istruzioni di salto, il cui carattere di funzione e' 0 (zero), che agiscono sul registro indirizzo istruzioni.

Al leggere una istruzione di salto il calcolatore deve :

a) esaminare se si e' o no verificata una determinata eventualita'. Questa viene indicata al calcolatore per mezzo del carattere p8, che serve dunque a distinguere un'istruzione di salto da una altra, e che in seguito sara' indicata con la lettera E. Se la eventualita' in esame non si e' verificata il calcolatore esegue l'istruzione immediatamente seguente l'istruzione di salto; si dice allora che il salto non e' stato eseguito;

b) se la eventualita' osservata si e' verificata, eseguire l'istruzione di indirizzo IIII (modificabile da T_m) indicato dalla istruzione di salto. Si dice allora che il salto e' stato eseguito;

c) ricordare, se necessario, l'indirizzo dell'istruzione di salto, per ritornare al programma da cui si e' saltati, dopo aver eseguito altre istruzioni. Tale necessita' si presenta quando una stessa sequenza di istruzioni (o sottoprogramma) puo' essere chiamata da diversi punti del programma principale. In p7 della istruzione di salto si deve allora indicare il registro T_s in cui si desidera venga ricordato l'indirizzo dell'istruzione stessa; in T_s viene registrato, per la precisione, l'indirizzo in memoria del carattere p8 di tale istruzione, e solo nel caso che il salto sia stato eseguito.

Hanno questa funzione le seguenti istruzioni :

SC=, SC ≠ , SC>, SC<, SR=, SR≠, SA=, SA>, SA<, SOM,

STO, SE_1, SE_2, SE_3, SE_4, SI, SIN, SN, STOP.

Salta se uguale da confronto		SC =
interno	(101001) X T_s I I I I T_m 0	10 o 15

a) Salta se nell'ultimo confronto, eseguito per conto del programma in cui la istru zione e' posta, la memoria e' risultata uguale al secondo termine di confronto.

Salta se diverso da confronto		SC ≠
interno	(101011) Y T_s I I I I T_m 0	10 o 15

a) Salta se nell'ultimo confronto, eseguito per conto del programma in cui la istru zione e' posta, la memoria e' risultata diversa dal secondo termine di confronto.

Salta se maggiore da confronto		SC >
interno	(100001) S' T_s I I I I T_m 0	10 o 15

a) Salta se nell'ultimo confronto, eseguito per conto del programma in cui la istruzione e' posta, la memoria e' risultata maggiore del secondo termine di confronto.

b) Nel caso di confronto fra due zone di memoria (operazioni PUM, CMM), salta se e' risultata maggiore la zona di memoria specificata dalla CMM.

Salta se minore da confronto		SC <
interno	(100011) T T_s I I I I T_m 0	10 o 15

a) Salta se nell'ultimo confronto, eseguito per conto del programma in cui la istruzione e' posta, la memoria e' risultata minore del secondo termine di confronto.

b) Nel caso di confronto fra due zone di memoria (operazioni PUM, CMM), salta se e' risultata minore la zona di memoria specificata dalla CMM.

Salta se il Risultato e' uguale a zero		SR =
interno	(100010) θ T_s I I I I T_m 0	10 o 15

a) Salta se e' uguale a zero il risultato dell'ultima istruzione, eseguita per conto del programma in cui l'istruzione di salto e' posta, del tipo +MA, +X, -MA, -X, MA, MT, +MT, -MT, XLD, XLN, +LD, +CT, -CT, CT, +MM, -MM, Y, +AM, +TM.

b) Per risultato di una istruzione MA, MT, CT, Y si intende l'informazione trasferita da dette istruzioni.

c) L'indicazione di risultato uguale a zero e' annullata, oltre che dalle precedenti istruzioni, anche dalla AoM e ToM.

Salta se il Risultato e' diverso da zero		SR ≠
interno	(100000) . T_s I I I I T_m 0	10 o 15

a) Salta se e' diverso da zero il risultato dell'ultima istruzione, eseguita per conto del programma in cui l'istruzione di salto e' posta, del tipo +MA, +X, -MA, -X, MA, MT, +MT, -MT, XLD, XLN, +LD, +CT, -CT, CT, +MM, -MM, Y, +AM, +TM.

b) Per risultato di una istruzione MA, MT, CT, Y si intende l' informazione trasferita da dette istruzioni.

c) L'indicazione di risultato diverso da zero e' annullata, oltre che dalle precedenti istruzioni, anche dalla AoM e ToM.

Salta se l'Accumulatore e' uguale a zero		SA= 0
interno	(100100) V T_s I I I I T_m 0	10 o 15

a) Salta se, a seguito dell'ultima istruzione eseguita del tipo +MA, +X, -MA, -X, MA, l'accumulatore e' risultato uguale a zero.

b) L'indicazione di accumulatore uguale a zero viene annullata quando si esegue una successiva istruzione del tipo anzidetto od una istruzione AoM.

Salta se l'Accumulatore e' maggiore di zero		SA > 0
interno	(100111) U T_s I I I I T_m 0	10 o 15

a) Salta se, a seguito dell'ultima istruzione eseguita del tipo +MA, +X, -MA, -X, MA, l'accumulatore e' risultato maggiore di zero.

b) L'indicazione di accumulatore maggiore di zero viene annullata quando si esegue una successiva istruzione del tipo anzidetto od una istruzione AoM.

Salta se l'Accumulatore e' minore di zero		SA < 0
interno	(101000) W T_s I I I I T_m 0	10 o 15

a) Salta se, a seguito dell'ultima istruzione eseguita del tipo +MA, +X, -MA, -X, MA, l'accumulatore e' risultato minore di zero.

b) L'indicazione di accumulatore minore di zero viene annullata quando si esegue una successiva istruzione del tipo anzidetto od una istruzione AoM.

Salta se vi e' stato Overflow in Memoria		SOM
interno	(101100) ' T_s I I I I T_m 0	10 o 15

a) Salta se si e' avuto overflow in memoria a seguito dell'ultima istruzione, eseguita per conto del programma in cui l'istruzione di salto e' posta, del tipo +MM, -MM, +AM, +TM

b) Le condizioni che provocano il segnale di overflow sono specificate istruzione per istruzione.

Salta se non vi e' stato Overflow in registro T		STO
interno	(101111) Z T_s I I I I T_m 0	10 o 15

a) Salta se non si e' avuto overflow nel registro T operato dalla ultima istruzione, eseguita per conto del programma in cui l'istruzione di salto e' posta, del tipo +MT, -MT, +CT, -CT.

b) Le condizioni che provocano il segnale di overflow sono specificate istruzione per istruzione.

Salta su condizione esterna 1		SE_1
interno	(111110) π T_s I I I I T_m 0	10 o 15

a) Salta su condizione esterna 1. L'esecuzione del salto e' subordinata al posizionamento della levetta 1 posta sul tavolo di comando. Il posizionamento viene effettuato manualmente dall'operatore.

Salta su condizione esterna 2		SE_2
interno	(111010) $ T_s I I I I T_m 0	10 o 15

a) Salta su condizione esterna 2 . Come SE_1 per levetta 2 .

Salta su condizione esterna 3		SE_3
interno	(100101) ! T_s I I I I T_m 0	10 o 15

a) Salta su condizione esterna 3 . Come SE_1 per levetta 3 .

Salta su condizione esterna 4		SE_4
interno	(101101) & T_s I I I I T_m 0	10 o 15

a) Salta su condizione esterna 4 . Come SE_1 per levetta 4 .

Non salta		SN
interno	(111100) R T_s X X X X T_m 0	10

a) Non salta mai, ma immagazzina nel registro T_s l'indirizzo del carattere p8 della istruzione.

b) Le posizioni p3 ÷ p6 non vengono prese in considerazione.

Salta incondizionatamente		SI
interno	(100110)) T_s I I I I T_m 0	15

a) Salta in qualsiasi caso.

Salto invariante		SIN
interno	(100110)) T_s C C C C T_s ~	19

a) Questa e' diversa dalle altre istruzioni di salto, in quanto non indica l'indirizzo della prossima istruzione da eseguire, ma l'addendo CCCC (non modificabile) da sommare al contenuto del registro indirizzo istruzioni.

b) Si noti che :

1) il suo carattere di Funzione non e' zero, ma ~ (approssimato);
2) non vi e' registro T_m modificatore; in p2 e p7 deve essere indicato il medesimo registro;
3) in T_s contrariamente agli altri salti viene immagazzinato l'indirizzo di p2 dell'istruzione.

c) Per effettuare un salto ad una istruzione registrata M istruzioni oltre la SIN, in CCCC si deve porre un numero uguale a :

$$(8 \cdot M) + 1$$

Dovendo ad esempio eseguire la terza istruzione oltre la SIN, si avra' :

$$CCCC = 0008 \cdot 3 + 0001 = 0025$$

d) Il registro T_s deve sempre essere indicato.

e) Il tempo di esecuzione e' di 19 periodi di cifre anziche' di 15.

A r r e s t o		STOP
interno	(111001) 0 X X X X X # 0	10

a) Pone termine allo svolgimento del programma nel quale l'istruzione e' posta.

b) Se l'istruzione e' posta nel 1° programma, questo ha termine, mentre lo svolgimento del 2° programma puo' proseguire.

c) Se si ha lo STOP nel 2° programma, questo ha termine mentre il 1° programma puo' proseguire.

d) L'istruzione STOP non arresta l'esecuzione della istruzione MS eventualmente in corso.

SCHEMA RIASSUNTIVO	Istruzioni di salto

Salti su confronti	:	SC=	SC≠	SC>	SC<
Salti su risultati	:	SR=	SR≠		
Salti su contenuto accum.	:	SA=	SA>	SA<	
Salti in overflow	:	SOM	STO		
Salti su condizioni est.	:	SE_1	SE_2	SE_3	SE_4
Salti speciali	:	SI	SN	SIN	STOP

Particolarita' dei salti

Funzione : alterare la successione delle istruzioni di programma se e' verificata una certa eventualita'.

Eventualita' : viene indicata all'elaboratore per mezzo del carattere p8 dell'istruzione di salto.

Registro T_s : ricorda l'indirizzo del carattere p8 dell'istruzione di salto solo nel caso che il salto venga eseguito.

Salta, se l'eventualita' si e' verificata, all'istruzione di indirizzo IIII indicato nell'istruzione di salto.
Tempo di esecuzione : 15 periodi di cifra.

Non salta, se l'eventualita' non si e' verificata, ed esegue l'istruzione immediatamente successiva del programma.
Tempo di esecuzione : 10 periodi di cifra.

CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE	CARATTERE DI FUNZIONE
SC=	interno	10 o 15	X T_s I I I I T_m	0
SC≠	"	10 o 15	Y T_s I I I I T_m	0
SC>	"	10 o 15	S T_s I I I I T_m	0
SC<	"	10 o 15	T T_s I I I I T_m	0
SR=	"	10 o 15	Θ T_s I I I I T_m	0
SR≠	"	10 o 15	• T_s I I I I T_m	0
SA=	"	10 o 15	V T_s I I I I T_m	0
SA>	"	10 o 15	U T_s I I I I T_m	0
SA<	"	10 o 15	W T_s I I I I T_m	0
SOM	"	10 o 15	' T_s I I I I T_m	0
STO	"	10 o 15	Z T_s I I I I T_m	0
SE_1	"	10 o 15	π T_s I I I I T_m	0
SE_2	"	10 o 15	$ T_s I I I I T_m	0
SE_3	"	10 o 15	! T_s I I I I T_m	0
SE_4	"	10 o 15	& T_s I I I I T_m	0
SI	"	15	) T_s I I I I T_m	0
SN	"	10	R T_s X X X X T_m	0
SIN	"	19	) T_s C C C C T_s	~
STOP	"	10	0 X X X X X #	0

7.10. Istruzioni di salto su errore

La probabilita' di un errore nel movimento di un ca-
rattere tra i vari organi dell'elaboratore e' estre-
mamente piccola.

E' tuttavia indispensabile cautelarci anche contro
questa evenienza con appositi dispositivi di macchi-
na che segnalino il verificarsi dell'errore.

Questi segnalatori nell'Elea 9003 esistono di fatto
ed i loro segnali sono rilevabili mediante apposite i
struzioni di salto.

Dette istruzioni inserite opportunamente nel program
ma permettono di ricorrere a programmi di correzione
ogni qualvolta si verifichi un errore nel corso di u
na elaborazione.

Tale errore evidentemente non puo' essere di natura
logica o imputabile ad inesattezze, a inversioni o
scambi di dati, o a difetti di quadratura, rilevabi-
li e correggibili con tecniche di altro genere.

Gli errori in questione possono essere imputati esclu
sivamente all'imperfetto funzionamento dell'elabora-
tore in un determinato caso.

Salta se errore qualsiasi		SE
interno	(110110) β T_s I I I I T_m 0	10 o 15

a) Salta se e' stato commesso un errore qualsiasi.

b) Oltre agli errori rilevati dai salti su errore del tipo SEMI, SEUM, SEME, SEA, SEU, SEN, SEL, rileva se vi e' stato errore nel lettore fotoelettrico di banda perforata, collegato al tavolo di comando.

c) Se l'istruzione SE rimanda ad una istruzione che non sia di salto, la segnalazione di errore qualsiasi viene annullata.

Salta se errore in uscita da memoria		SEUM
interno	(110000) ε T_s I I I I T_m 0	10 o 15

a) Salta se e' stato rilevato un errore nell'estrazione dalla memoria attraverso il canale interno.

b) La rilevazione dell'errore avviene per mezzo del controllo di disparita'.

Salta se errore in memoria, canale interno		SEMI
interno	(110010) * T_s I I I I T_m 0	10 o 15

a) Salta se e' stato rilevato un errore nella introduzione in memoria attraverso il canale interno.

b) La rilevazione dell'errore avviene per mezzo del controllo di disparita'.

Salta se errore in memoria, canale esterno		SEME
interno	(110001) J T_s I I I I T_m 0	10 o 15

a) Salta se e' stato rilevato un errore nella introduzione in memoria attraverso il canale esterno.

b) La rilevazione dell'errore avviene per mezzo del controllo di disparita'.

Salta se errore in accumulatore		SEA
interno	(110011) K T_s I I I I T_m 0	10 o 15

a) Salta se e' stato rilevato un errore nella estrazione dall'accumulatore.

b) La rilevazione avviene per mezzo del controllo di disparita'.

Salta se errore in unita' aritmetica		SEU
interno	(110111) L T_s I I I I T_m 0	10 o 15

a) Salta se l'unita' aritmetica e logica ha commesso un errore.

b) La rilevazione puo' avvenire per mezzo :
   1) del controllo di disparita';
   2) della prova del 3 ;
   3) o dalla presenza di configurazioni speciali in operazioni aritmetiche.

Salta se errore in unita' a nastro magnetico		SEN
interno, GUN	(110100) M T_s I I I I T_m 0	10 o 15

a) Salta se una unita' a nastro magnetico ha commesso un errore.

b) La rilevazione avviene per mezzo del controllo di disparita'.

c) Per le unita' che registrano in avanti la rilevazione avviene inoltre per mezzo del controllo di registrazione per rilettura e confronto.

d) L'istruzione non viene eseguita se e' in corso una eventuale operazione di nastro.

SCHEMA RIASSUNTIVO	Salti su errori

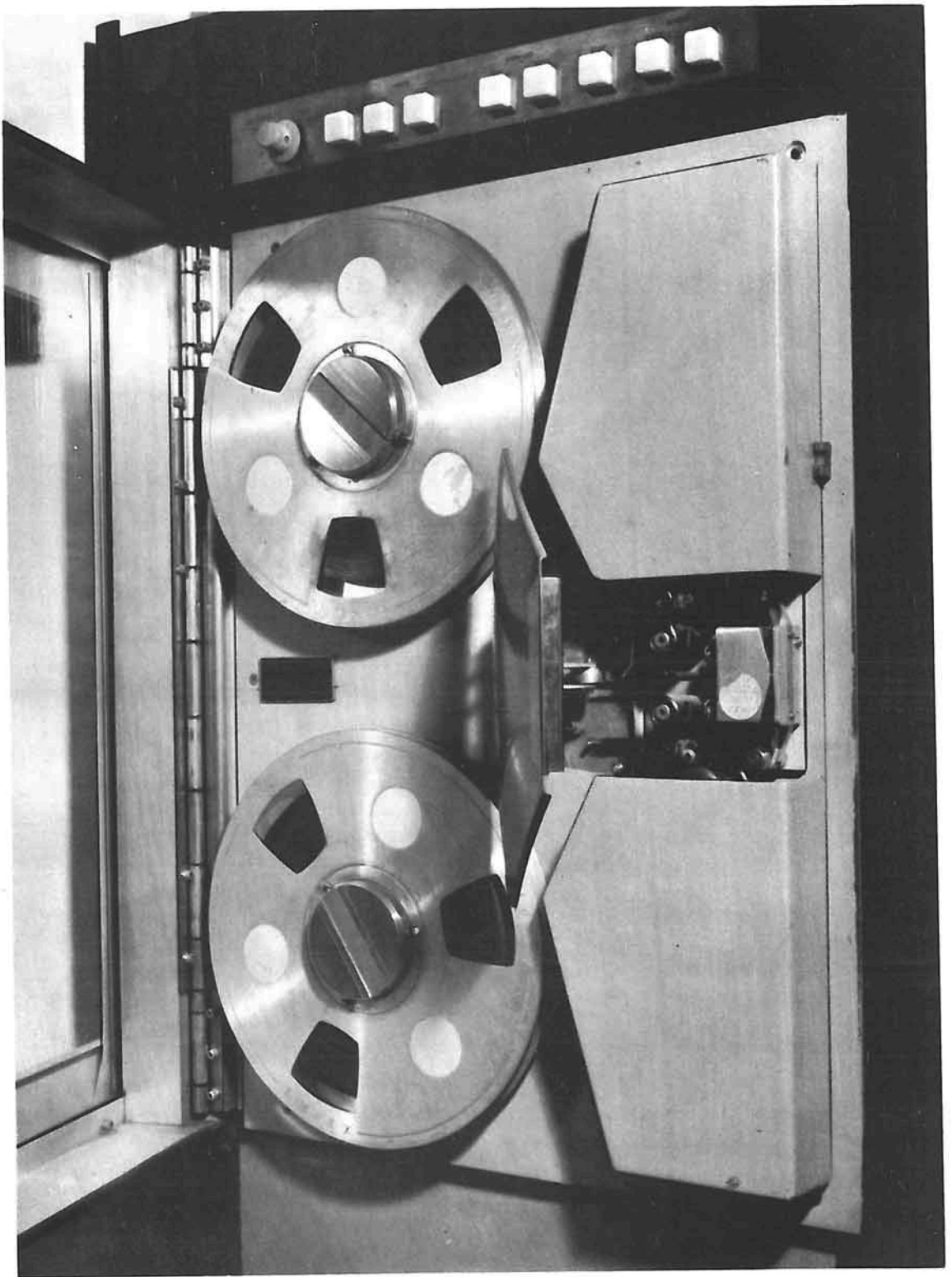
Istruzioni : SE SEMI SEUM SEME SEA SEU SEN SEL

Particolarita' dei salti su errori

Funzione : controllo della corretta elaborazione delle informazioni; il salto viene eseguito se lo errore si e' verificato.

Caratteristiche : l'indicazione di un tipo di errore viene conservata dall'istante in cui l'errore si e' rilevato fino alla fine della esecuzione dell'istruzione di salto di quel tipo di errore, oppure fino alla esecuzione di una istruzione di salto relativa a piu' tipi di errore.

I I I I : che compare in queste istruzioni e' generalmente l'indirizzo della prima istruzione di un sottoprogramma di segnalazione di errore o di correzione di errore. Per le istruzioni SCA e SE la prima istruzione del sottoprogramma elaborativo deve essere necessariamente un'istruzione di salto.

CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE	CARATTERE DI FUNZIONE
SE	interno	10 o 15	β T_s I I I I T_m	0
SEMI	"	10 o 15	* T_s I I I I T_m	0
SEUM	"	10 o 15	ε T_s I I I I T_m	0
SEME	"	10 o 15	J T_s I I I I T_m	0
SEA	"	10 o 15	K T_s I I I I T_m	0
SEU	"	10 o 15	L T_s I I I I T_m	0
SEN	int. GUN	10 o 15	M T_s I I I I T_m	0
SEL	int. unita' linea U	10 o 15	γ U I I I I T_m	0

UNITA' A NASTRO MAGNETICO FR - 400 .-

CAP. 8° : UTILIZZAZIONE DEI NASTRI MAGNETICI

8.1. Caratteristiche generali

Il nastro magnetico come supporto d'informazione per l'elaborazione elettronica dei dati e' certamente il piu' funzionale : e' veloce, facilmente trasferibile, garantisce la buona conservazione dei dati ed e' poco costoso.

Puo' essere usato un numero illimitato di volte sia per la lettura dei dati in esso immagazzinati che per la registrazione di nuove informazioni su quelle preesistenti.

Viene avvolto su bobine che possono essere conservate col minimo ingombro portando ad economia di spazio molto rilevanti.

La durata di una registrazione e' in pratica illimitata, e i dati sono molto meno soggetti ad alterazioni che su altri supporti.

Le operazioni eseguibili su nastro magnetico sono :

la registrazione di informazioni,
la lettura delle informazioni gia' registrate.

L'esecuzione di dette operazioni e' affidata ad una apposita apparecchiatura chiamata "unita' a nastro".

Le unita' a nastro sono dotate di due gruppi di testine magnetiche rispettivamente per la lettura e la registrazione, e dei dispositivi di segnalazione di fine nastro e di arresto delle bobine.

Dall'abbinamento delle caratteristiche tecniche e funzionali dei nastri e delle relative unita' dipen

dono i seguenti requisiti :

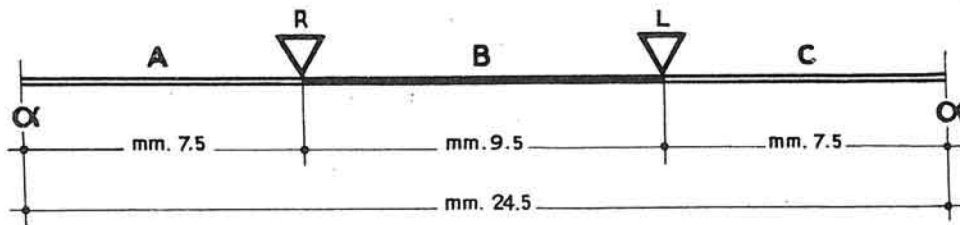
- la velocita' di trascinamento del nastro magnetico
- la densita' di registrazione
- la frequenza di registrazione.

Considerando che su una zona di circa 2,54 centimetri sono registrabili 300 caratteri, e che la velocita' di un nastro puo' andare oltre i 370 centimetri al secondo, appare evidente come sia impossibile arrestare un nastro sotto le testine di lettura e registrazione tra due specifici caratteri.

Si deve inoltre osservare che sia la lettura che la registrazione su nastro magnetico sono possibili solo quando esso funzioni a velocita' di regime; tale velocita' pero' non e' raggiungibile istantaneamente dallo stato di fermo, ne' il nastro funzionante a velocita' di regime puo' essere bloccato all'istante a causa della forza d'inerzia contrapposta dalla bobina.

Si e' reso pertanto necessario il raggruppamento delle informazioni registrate su nastro in blocchi separati da una piccola zona non registrata chiamata "interblocco".

Questo intervallo consente al nastro di avviarsi e di raggiungere la velocita' di regime quando la testina di lettura si posiziona sul primo carattere del blocco, oppure, una volta letto l'ultimo carattere del blocco di posizionare le testine al centro dell'interblocco stesso.

Questo intervallo di circa 2,5 cm e' diviso in tre zone :

dove A e C indicano le zone occorrenti per l'avvio o l'arresto del nastro, e B indica la zona riservata per il posizionamento delle testine.

In linea di massima si puo' affermare che quanto piu' lunghi sono i blocchi, tanto minori saranno le zone di nastro e il tempo perduto per gli avvii e gli arresti.

Il dimensionamento dei blocchi viene effettuato generalmente in rapporto alla capacita' della memoria di lavoro dell'elaboratore; ovviamente una memoria di capacita' ridotta implichera' l'impiego di blocchi di piccole dimensioni che esalteranno gli inconvenienti citati.

8.2. La registrazione e la lettura del nastro

I caratteri rappresentati nella memoria di lavoro mediante sei bit vengono riprodotti sul nastro magnetico in modo analogo, con la sola variante che in luogo di nuclei magnetici si hanno aree magnetizzabili.

La registrazione, che avviene sempre in avanti, e' sottoposta a un duplice controllo :

- un primo controllo e' quello di disparita', dato da un settimo bit che viene registrato da un'ap-

posita testina in parallelo con i sei bit raffigu ranti il carattere.
Mediante tale controllo si verifica che non vi sia perdita o generazione spuria di un bit; infatti il numero delle aree effettivamente magnetizzate com presa quella di controllo, deve essere sempre di- spari per ogni carattere;

- un secondo controllo e' ottenuto mediante la let- tura e l'immediato confronto di quanto e' appena registrato con la informazione originaria.

Risulta evidente l'assoluta garanzia che ne deriva circa una corretta registrazione dei dati che si vo- gliono temporaneamente conservare in archivio per successive elaborazioni.

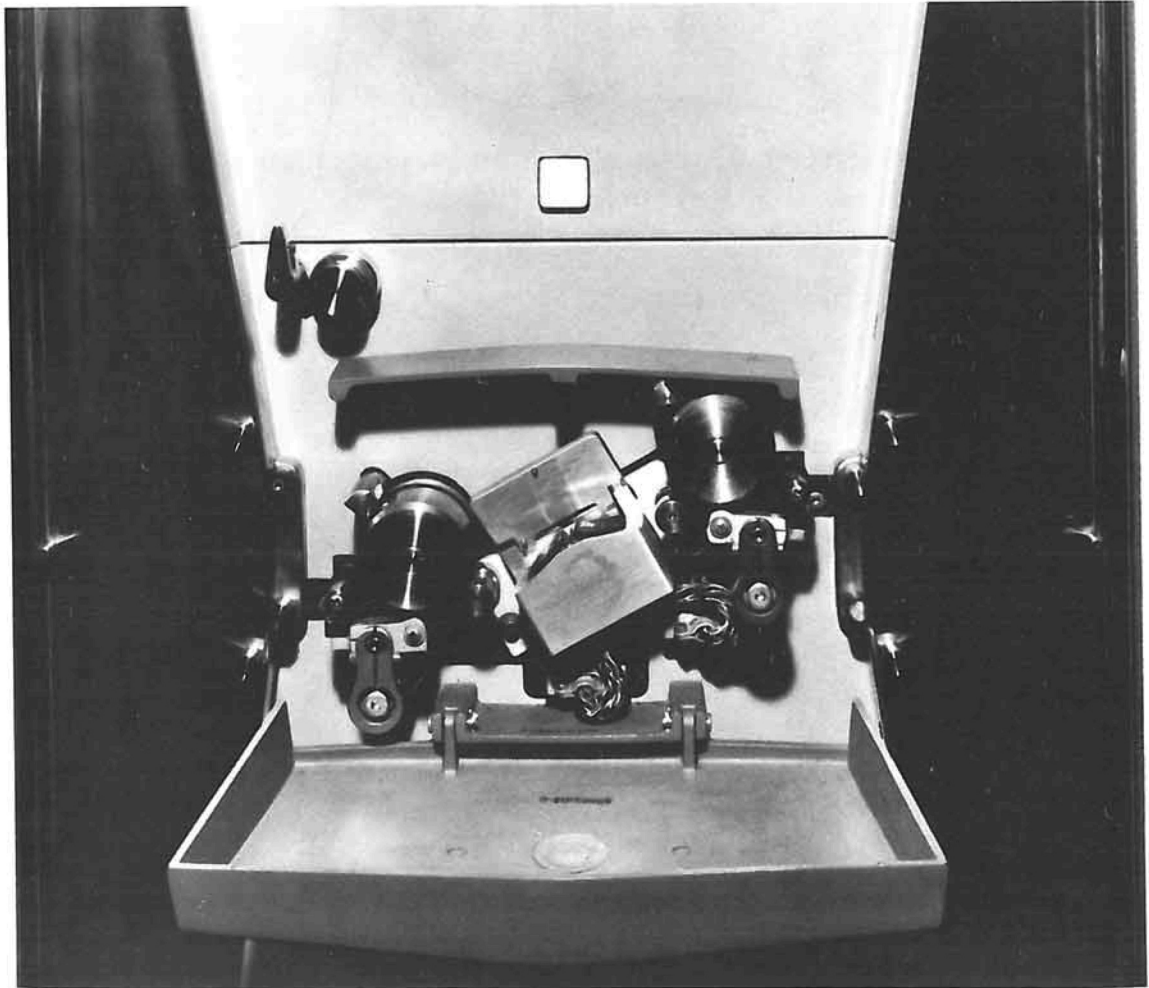
Nel caso fosse riscontrato un errore per mezzo dei suddetti controlli, esiste la possibilita' di appor tare la necessaria correzione prima che la registra zione prosegua.

Diversamente dalla registrazione la lettura puo' av venire sia avanti che indietro, ed il controllo e' ottenuto solo mediante la verifica del bit di dispa rita'.

Per scandire i tempi di lettura in corrispondenza di ogni carattere esiste un bit supplementare chiamato "bit di temporizzazione" che provoca la lettura del la zona di nastro in cui esso e' posto.

Come gia' si e' accennato, le unita' a nastro magne tico sono collegate all'elaboratore tramite uno spe cifico governo : governo unita' nastro (GUN).

Il GUN e' dotato di tre memorie :

a) la memoria di ricerca, di 128 posizioni, che con- tiene la chiave di ricerca per l'individuazione di uno specifico blocco di informazioni.

DISPOSITIVO DI LETTURA E REGISTRAZIONE FR - 300 .—

Mediante questo accorgimento tecnologico, e' possibile infatti raggiungere qualsivoglia blocco registrato su di un nastro ed apporvi le modifiche o gli aggiornamenti richiesti dai particolari problemi di lavoro;

b) la memoria di controllo della registrazione, di 256 caratteri, che ha la funzione di contenere i caratteri registrati ma non ancora riletti;

c) la memoria di trascrizione, la cui capacita' e' di 2048 posizioni.
Quest'ultima memoria permette di eliminare eventuali differenze d'impaccamento causate dalle diverse condizioni tecnologiche di funzionamento eventualmente verificatesi in successive operazioni di registrazione, e da modo inoltre di evitare che in operazioni di trascrizione da nastro a nastro si sommino tra di loro successive differenze di frequenza di impaccamento.

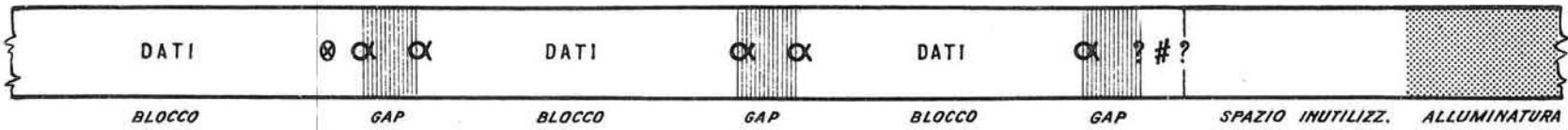
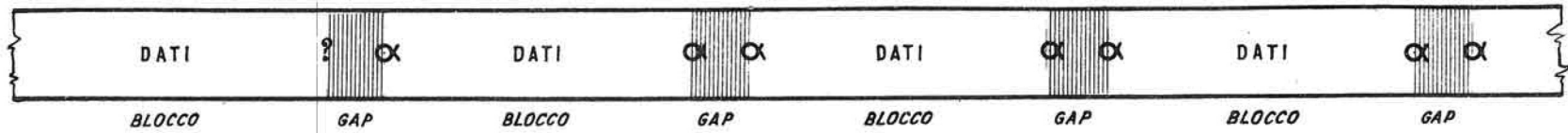
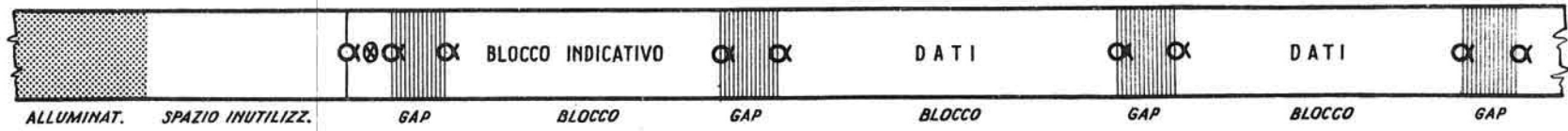
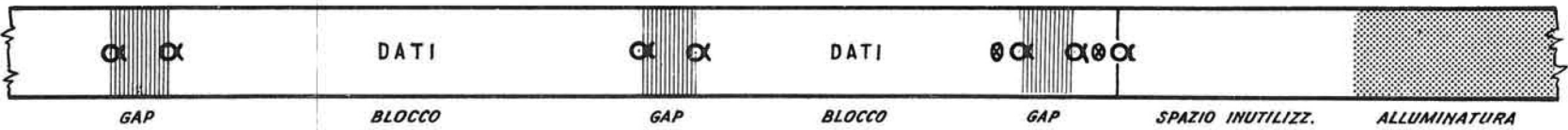
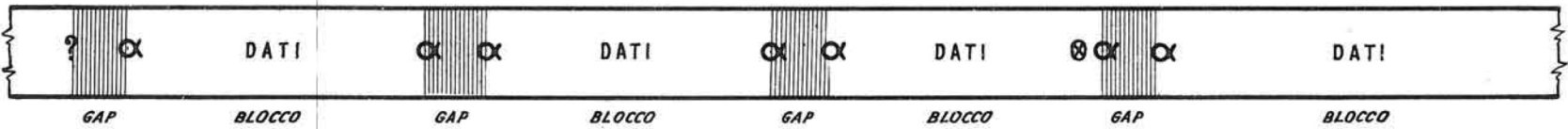
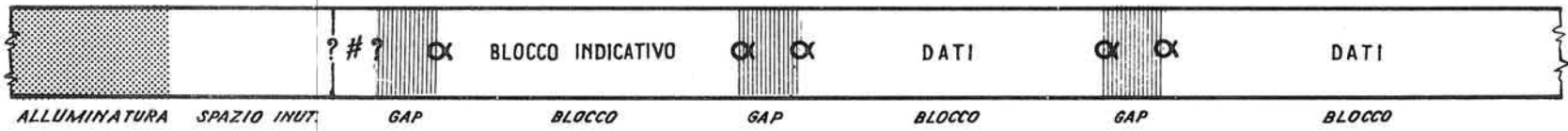
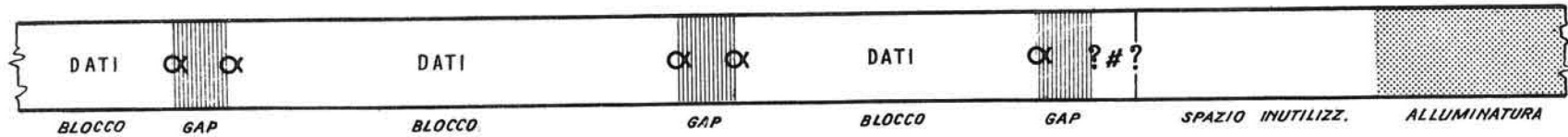
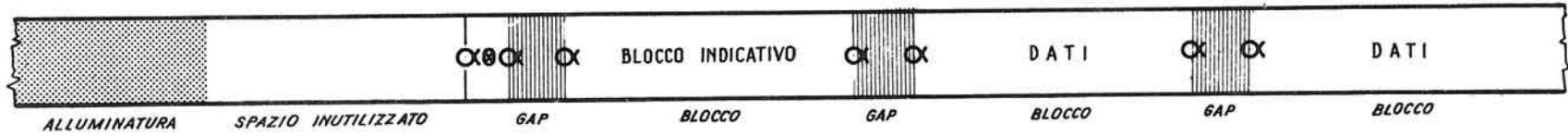
8.3. Organizzazione delle informazioni su nastro magnetico

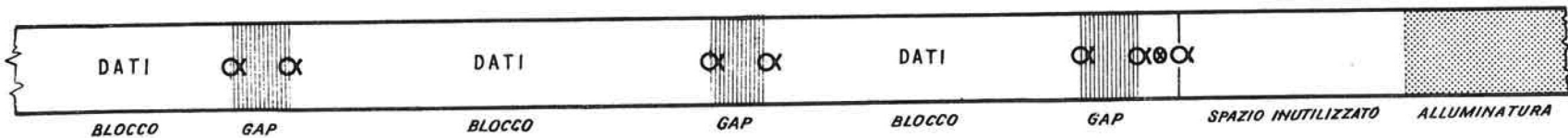
Gruppi di caratteri, a seconda della funzione per la quale sono stati creati, possono costituire le seguenti unita' di elaborazione :

1° la parola
2° la scheda
3° il blocco
4° la sequenza
5° l'informazione

La parola e' l'elemento minimo elaborabile e costituisce un dato.

La scheda comprende una o piu' parole e costituisce l'insieme dei dati relativi a un documento.

ORGANIZZAZIONE SU NASTRO - 1º metodo
NASTRO Nº 1
BLOCCO INDICATIVO
DATI
ALLUMINAT.
SPAZIO INUTILIZZ.
GAP
BLOCCO
ALLUMINATURA
NASTRO Nº 2
SPAZIO INUT.

ORGANIZZAZIONE SU NASTRO - 2º metodo

NASTRO Nº 1

NASTRO Nº 2

Il blocco e' costituito da un numero variabile di schede ed e' creato in funzione delle caratteristiche tecnologiche del sistema.

La sequenza e' un insieme di blocchi omogenei.

L'informazione comprende tutte le sequenze relative ad un unico flusso di elaborazione.

Questi elementi sono individuabili sul nastro attraverso speciali caratteri che li delimitano.

La parola — fa eccezione alla regola in quanto e' reperibile anche senza speciali caratteri di inizio e fine.

La scheda — termina col carattere teta (θ) solo quando viene operata mediante la particolare istruzione NDN.

Il blocco — inizia e termina col carattere alfa (α).

La sequenza — termina col carattere punto interrogativo che sostituisce uno dei due alfa che delimitano l'ultimo blocco della sequenza.

La informazione — inizia e termina coi caratteri abbinati "moltiplicato per" ($\otimes$) ed alfa (α).

Nella creazione dei blocchi si devono inoltre osservare le seguenti norme :

1. La lunghezza dei blocchi in trascrizione deve essere al minimo di 3 caratteri inclusi i caratteri di inizio e fine blocco.
   Per esempio sono blocchi accettabili :

   $\alpha \otimes \alpha$ oppure α A ?

2. Per quanto riguarda l'istruzione NDN i blocchi di nastro magnetico devono avere come lunghezza minima 10 caratteri e come lunghezza massima 5000 caratteri.

3. Per quanto riguarda l'istruzione TN la lunghezza dei blocchi puo' essere al massimo di 5000 caratteri.

NB

a) Nella casella N^{o} 5 e' indicato solo il tempo necessario alla fase preparatoria; la durata della fase esecutiva dipende infatti dal numero di caratteri o di blocchi che si vogliono operare.

b) Le segnalazioni di macchina relative alle istruzioni SFS, SFI, SN_1, SN_s, vengono annullate ad avvenuta lettura dell'istruzione interessata.

Leggi Nastro, avanti		LNa
esterno, GUN	X n_1 I I I I T_m * (110010)	10

X : posizione non utilizzata

n_1 : nome del nastro in lettura

I I I I : indirizzo di memoria a partire dal quale vengono registrati i caratteri da nastro

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del nastro n_1

* : legge dal nastro montato sull'unita' n_1 (specificata in p7) alla Memoria, procedendo in avanti

a) Il trasferimento avviene dal segnale di inizio blocco (α , ?) al segnale di fine blocco (α , ?).

b) I caratteri di inizio e fine blocco (α , ?) vengono trasferiti in memoria.

c) I caratteri letti sul nastro vengono registrati in Memoria a partire dall'indirizzo IIII e per indirizzi crescenti.

Leggi Nastro, indietro		LNi
esterno, GUN	X n_1 I I I I T_m K (110011)	10

X : posizione non utilizzata

n_1 : nome del nastro in lettura

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del nastro n_1

K : legge dal nastro montato sull'unita' n_1 (specificata in p7) alla Memoria, procedendo indietro

a) Come la LNa, ma con senso di marcia indietro del nastro e per indirizzi decrescenti.

Registra Nastro		RN
esterno, GUN	X n_s I I I I T_m M (110100)	10

X : posizione non utilizzata

n_s : nome del nastro in scrittura

I I I I : indirizzo di memoria a partire dal quale vengono registrati i caratteri

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del nastro n_s

M : registra dalla Memoria, sul nastro montato sull'unita' n_s (specificata in p7), procedendo in avanti

a) Il trasferimento ha inizio dall'indirizzo IIII della Memoria e prosegue fino al segnale di fine blocco (α ; ?'), per indirizzi crescenti.

b) All'indirizzo IIII deve essere registrato il segnale di inizio blocco (α , ?).

c) Il blocco da registrare deve essere di almeno tre caratteri, inclusi i segnali di inizio e fine blocco.

Prepara Ricerca Nastri			PRN
esterno, GUN	X X I I I I T_m 0 (111001)		10

X X : posizioni non utilizzate

I I I I : indirizzo di memoria a partire dal quale si trasferisce alla memoria del GUN

T_m : il registro T_m puo' modificare l'indirizzo IIII

0 : trasferisce dalla memoria dell'unita' centrale, a partire dall'indirizzo IIII e per indirizzi crescenti, alla memoria del GUN, i caratteri comprendenti la parola chiave che sara' usata nella ricerca su nastro.

a) La ricerca puo' avvenire con una parola chiave avente al massimo la lunghezza di 128 caratteri; non e' necessario che i caratteri interessati siano adiacenti purche' le posizioni non interessate alla ricerca siano diesizzate; il trasferimento della parola chiave dalla memoria al GUN si arresta quando viene letto un ⊠ dopo un carattere diverso da ⊠ .

b) Il carattere ("⊠" o "?") che obbligatoriamente inizia ogni blocco di nastro, non deve essere considerato; nel senso che la parola chiave non deve contenerlo, ne' deve contenere al suo posto un carattere # .

c) Se si desidera che il confronto avvenga con i caratteri registrati su nastro dalla $(n+1)^a$ alla $(n+m)^a$ posizione di ogni blocco $(n+m < 128)$ la parola da trasferire dovra' comprendere inizialmente n caratteri # quindi la parola chiave di m caratteri, e finalmente il carattere ⊠ .

Esempio : per $n = 4$ $m = 6$

# # # # 4 8 7 2 3 1 ⊠	impostazione parola chiave da trasferire nel GUN .
A B C D 4 8 7 2 3 1 N	organizzazione del nastro nella zona corrispondente alla parola chiave.

d) Il blocco ricercato sara' quello contenente la parola chiave trasferita nel GUN; o, in mancanza di questo, il blocco contenente una parola chiave di valore superiore a quella trasferita.

e) Se nella ricerca la macchina non incontrasse alcuno di tali blocchi, per provocare l'arresto dell'unita' a nastro sara' necessario prevedere sul nastro stesso un ultimo blocco fittizio contenente tanti ⊠ quanti sono i caratteri che esprimono l'intero codice.

Esempio :

parola chiave	# # 1 5 7 9 ⊠
blocco fittizio	⊠ ⊠ ⊠ ⊠ ⊠ ⊠ ⊠

f) La ricerca e' convalidata per tutti i caratteri che nella parola trasferita dalla PRN sono diversi dal carattere " # " ; si rende quindi possibile la ricerca automatica di blocchi con chiave spezzata.

Esempio : # # # 2 5 7 # # 3 5 9 # A ⊠

Trascrivi Nastro		TN
GUN	n_l n_s B B B B T_m $ (111010)	10

n_l : nome del nastro in lettura

n_s : nome del nastro in scrittura

B B B B : numero di blocchi da trasferire da nastro a nastro

T_m : il registro T_m puo' modificare sia il numero BBBB che il nome del nastro n_s

$: trascrive dall'unita' a nastro n_l alla unita' a nastro n_s un numero BBBB di blocchi.

a) Se la TN e' immediatamente preceduta da una PRN si ha che :
   1) nelle posizioni p3 ÷ p6 e' registrato il carattere x ;
   2) l'istruzione trascrive da nastro a nastro un numero imprecisato di blocchi , l'ultimo dei quali e' quello che nelle sue posizioni iniziali contiene una parola uguale o maggiore di quella trasferita nella memoria del GUN per mezzo della PRN.

b) Con unita' lettura inesistente, codificata # n_s # ## # # $, il nastro dell'unita' in registrazione viene cancellato totalmente.

c) Se durante la trascrizione dei BBBB blocchi si verifica un errore di registrazione la macchina ne da' segnalazione e si arresta nel blocco dove si e' verificato l'errore stesso.

d) La lunghezza di un blocco non puo' superare i 5000 caratteri.

Prepara Ingresso da Nastro		PIN
est, int, GUN	X X I I I I T_m R (111100)	10

X X : posizioni non utilizzate

I I I I : indirizzo di memoria a partire dal quale si registra da n_l (specificato nell'istruzione NAN seguente)

T_m : il registro T_m puo' modificare l'indirizzo IIII

R : fissa l'indirizzo di memoria a partire dal quale vengono registrati i caratteri letti sul nastro n_l

a) L'istruzione PIN deve precedere immediatamente la NAN.

da Nastro a Nastro NO		NAN
est, int, GUN	n_l n_s I I I I T_m ε (110000)	10

n_l : nome del nastro in lettura

n_s : nome del nastro in registrazione

I I I I : indirizzo di memoria a partire dal quale si registra su n_s

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del nastro n_s

ε : registra dalla memoria su nastro montato sull'unita' n_s (specificata in p7) e simultaneamente legge dal nastro montato sull'unita' n_l (specificata in p8) su una diversa zona di memoria.

a) Ciascun carattere letto in memoria a partire dall'indirizzo IIII specificato dalla NAN viene trasferito sul nastro n_s; ciascun carattere letto dal nastro n_l viene registrato in memoria a partire dall'indirizzo IIII specificato dalla PIN immediatamente precedente.
b) Se l'indirizzo specificato dalla PIN e' uguale o maggiore di quello indicato dalla NAN, puo' accadere che i caratteri prelevati da n_l vadano a sostituire in memoria i caratteri che avrebbero dovuto essere piu' tardi registrati su n_s.
c) Con le istruzioni PIN-NAN possono essere operati blocchi di lunghezza variabile.
d) I due nastri funzionano in avanti; l'istruzione ha termine quando sono stati letti (sul nastro n_l e in memoria) entrambi i caratteri di fine blocco.
e) Non e' possibile porre n_s o n_l inesistenti (carattere #) .

NO

Da Nastro, seguendo una Direttrice a Nastro		NDN
est, int, GUN	n_1 n_s I I I I T_m β (110110)	10 + 1

n_1 : nome del nastro in lettura

n_s : nome del nastro in registrazione

I I I I : indirizzo della direttrice

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del nastro n_s

β : registra dalla memoria su nastro montato sull'unita' n_s e simultaneamente legge dal nastro montato sull'unita' n_1 sulle stesse zone di memoria.

a) Ciascun carattere letto da nastro va ad occupare la posizione di memoria da cui si e' immediatamente prima prelevato il carattere da registrare.

b) Le informazioni vengono lette e registrate a gruppi di caratteri la cui fine e' contraddistinta dal carattere "θ" per es. :

1 2 3 4 5 6 7 8 9 θ

c) I blocchi ed i gruppi corrispondenti devono avere la stessa lunghezza nei due nastri n_s e n_1 :

αxxxx$\theta\theta$xx$\theta\theta$xxx$\theta\theta$xx$\theta\theta$xxxxxα

αxxxx$\theta\theta$xx$\theta\theta$xxx$\theta\theta$xx$\theta\theta$xxxxxα

d) La lunghezza di un blocco organizzato per essere operato con la NDN puo' variare da 10 a 5.000 caratteri.

e) Il primo gruppo e' costituito dal carattere di inizio blocco, seguito immediatamente o dal primo carattere "θ" ed e' quindi composto di due caratteri : "$\alpha\theta$"; oppure dal contenuto della prima informazione ed e' allora composto da un numero di caratteri pari a quelli dell'informazione + 1.

f) In memoria, a partire dall'indirizzo IIII (modificabile da T_m) e per indirizzi decrescenti, e' registrata una lista consecutiva di indirizzi di memoria chiamata "direttrice".

1) Ognuno di questi indirizzi e' l'indirizzo di memoria a partire dal quale, per indirizzi crescenti, vengono prelevati e sostituiti i caratteri costituenti un gruppo.

2) Il primo indirizzo di destra della direttrice e' l'indirizzo del gruppo contenente l'α di inizio blocco, mentre l'ultimo indirizzo di sinistra corrisponde o all'indirizzo dell'α di fine blocco o all'indirizzo di sinistra del gruppo che lo contiene.

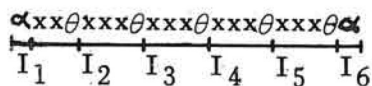
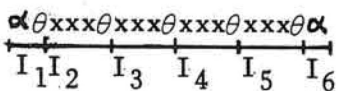
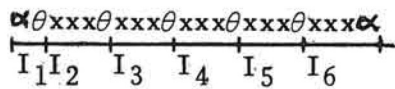
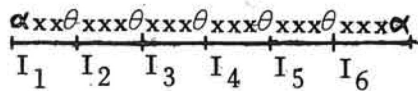
Il numero (N) degli indirizzi di una direttrice puo' essere quindi costituito :

1) dal numero delle informazioni costituenti il blocco : N

2) dal numero delle informazioni costituenti il blocco piu' l'indirizzo del gruppo "$\alpha\theta$" di inizio blocco : N + 1

./.

3) dal numero delle informazioni costituenti il blocco più' l'indirizzo dell'α di fine blocco : N + 1

4) dal numero delle informazioni costituenti il blocco più' l'indirizzo del gruppo "α θ" di inizio blocco e l'indirizzo dell'α di fine blocco : N + 2

g) I due nastri funzionano in avanti : l'istruzione ha termine quando viene letto il carattere di fine blocco.

h) Ponendo n_s inesistente (carattere ≠) si esegue solo il trasferimento da n_1 verso memoria secondo l'ordine stabilito dalla direttrice.
Analogamente, ponendo n_1 inesistente (carattere ≠) si esegue solo il trasferimento da memoria verso n_s secondo l'ordine stabilito dalla direttrice.

NASTRO IN REGISTRAZIONE
α ⊖ ⊖ A ⊖ ⊖ B ⊖ ⊖ C ⊖ ⊖ D ⊖ ⊖ E ⊖ ⊖ F ⊖ α
MEMORIA PRINCIPALE
DIRETTRICE
α⊖
A
B
C
D
E
F
α
α ⊖
A ⊖
B ⊖
C ⊖
D ⊖
E ⊖
F ⊖
α
NASTRO IN LETTURA
α ⊖ ⊖ A ⊖ ⊖ B ⊖ ⊖ C ⊖ ⊖ D ⊖ ⊖ E ⊖ ⊖ F ⊖ α

Disponi Unita' e Blocco		DUB
GUN	n J B B B B T_m N (111000)	10

n	:	nome del nastro interessato
J	:	indicazione del senso di marcia
B B B B	:	numero dei blocchi del nastro interessati dall'istruzione
T_m	:	il registro T_m puo' modificare sia il numero BBBB che il carattere indicato in J
N	:	fa procedere il nastro montato sull'unita' n (specificata in p8) del numero di blocchi indicato in BBBB, in avanti o all'indietro, secondo il carattere indicato in J.

a) Nelle posizioni da p3 a p6 non e' indicato un indirizzo di memoria.

b) Se nel carattere indicato in J il bit "a" e' zero, il numero di blocchi e' uguale a BBBB; se invece e' "1" il numero di blocchi e' 10.000 + BBBB.

c) Se nel carattere indicato in J il bit "d" e' zero, lo svolgimento avviene in avanti ; se invece e' "1" avviene all'indietro.

d) Se durante l'esecuzione della presenta istruzione si verifica un errore, la macchina ne da indicazione e procede fino alla fine della conta dei blocchi.

Cancella nastro		KN
esterno, GUN	X n I I I I T_m P (111011)	10

X : posizione non utilizzata

n : unita' a nastro interessata

I I I I : indirizzo di memoria che determina l'inizio dell'operazione

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del nastro n

P : cancella il nastro montato sull'unita' n, per una lunghezza determinabile in pollici

a) La cancellazione ha luogo durante il tempo impiegato dalla macchina a percorrere le posizioni di memoria che vanno dall'indirizzo IIII fino al primo carattere α (alfa).

b) La porzione di nastro che ne risulta cancellata e' determinabile con la formula $L = N \cdot v/f$ dove :
L = lunghezza nastro cancellato, in pollici;
N = numero caratteri intercorrenti fra l'indirizzo IIII ed il primo carattere α (alfa);
v = velocita' del nastro in pollici per secondo;
f = frequenza di lettura della memoria, in caratteri al secondo.

c) La KN viene normalmente utilizzata per cancellare una porzione di nastro.

d) Il contenuto della memoria rimane inalterato. All'indirizzo IIII non deve essere registrato il carattere α (alfa).

e) Per comodita' di calcolo indichiamo una formula approssimata per determinare lo spazio di nastro cancellato :

$$L = 2{,}1 + N \cdot 1{,}98 \cdot 10^{-3}$$

dove N e' il numero di caratteri intercorrenti tra l'indirizzo IIII ed il primo α ; L e' espresso in centimetri, con una approssimazione di $\pm$ 1/2 centimetro.

Avvolgi Nastro		AV
unita' a nastro	X n X X X X T_m ← (001101)	10

X X X X X : le posizioni p3, p4, p5, p6, p8 non sono utilizzate

n : nome dell'unita' a nastro interessata

T_m : il registro T_m puo' modificare il nome dell'unita' a nastro n

← : riavvolge all'indietro la bobina montata sull'unita' a nastro n fino al termine del nastro.

a) L'operazione di riavvolgimento non impegna il GUN ma solo l'unita' interessata, e' pertanto possibile far eseguire simultaneamente :
   1) piu' operazioni di riavvolgimento purche' interessino unita' diverse;
   2) una qualsiasi delle istruzioni relative ai nastri, di tipo LN, NDN, PRN, TN ecc. purche' non interessino l'unita' n .

b) Durante l'operazione di riavvolgimento :
   1) una istruzione riguardante la stessa unita' resta bloccata fino a terminata esecuzione dell'AV ;
   2) l'eventuale istruzione SNO relativa alla stessa unita' viene eseguita.

c) Se il GUN e' impegnato in una normale operazione, non e' in grado di ricevere istruzioni di riavvolgimento.

Salta se fine sequenza		SFS
interno	(111000) N T_s I I I I T_m 0	10 o 15

a) Salta se l'ultimo blocco letto da nastro inizia o termina con il carattere "?" di fine sequenza.

b) L'istruzione non viene eseguita se e' in corso un'eventuale operazione di nastro.

c) La segnalazione relativa a questa istruzione puo' essere annullata oltre che dalla lettura dell'istruzione stessa anche dalla lettura di una istruzione qualsiasi di nastro.

Salta se fine nastro in lettura		SN_l
interno	(111011) P T_s I I I I T_m 0	10 o 15

a) Salta se e' terminato il nastro in lettura.

b) L'istruzione non viene eseguita se e' in corso un'eventuale operazione di nastro.

Salta se fine nastro in registrazione		SN_s
interno	(111111) Q T_s I I I I T_m 0	10 o 15

a) Salta se e' terminato il nastro in registrazione.

b) L'istruzione non viene eseguita se e' in corso un'eventuale operazione di nastro.

Salta se fine informazione		SFI
interno	(111101) η T_s I I I I T_m 0	10 o 15

a) Salta se l'ultimo blocco da nastro termina con il carattere di fine informazione " Ø " .

b) L'istruzione non viene eseguita se e' in corso una eventuale operazione di nastro.

c) La segnalazione relativa a questa istruzione puo' essere annullata oltre che dalla lettura dell'istruzione stessa anche dalla lettura di una istruzione qualsiasi di nastro.

Salta se nastro occupato		SNO
interno	(------) n T_s I I I I T_m 0	10 o 15

a) Salta se l'unita' a nastro "n" e' gia' impegnata.

b) Il carattere di eventualita' non e' fisso, ma e' il carattere che contraddistingue l'unita' a nastro interessata.

c) La segnalazione di nastro occupato viene annullata dalla lettura del primo salto incontrato dopo l'esecuzione della presente istruzione.

Salta su Condizione Automatica		SCA
interno	(110101) (T_s I I I I T_m 0	10 o 15

(: carattere di eventualita'

T_s : registro che ricorda l'indirizzo della posizione p8 dell'istruzione

I I I I : indirizzo dell'istruzione alla quale si deve saltare

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro T_s

0 : salta se si e' verificata una delle seguenti eventualita' : riconoscimento di carattere chiave, unita' nastro non disponibile, condizione d'errore o condizione esterna SE_4.

a) L'analisi di quale condizione si e' verificata va eseguita mediante un sottoprogramma con il quale si possono ricavare le indicazioni particolari relative a ciascun organo e al tipo di segnalazione da esso rilevata.

b) Generalmente l'istruzione SCA viene posta di seguito a una istruzione di nastro di tipo LN, NDN, -NAN.

c) La sua esecuzione dipende dall'impegno del canale interno : se questo e' libero la SCA viene eseguita, altrimenti resta bloccata.

d) Per la NDN e PIN -NAN la SCA non viene eseguita fino ad ultimata operazione di lettura e registrazione, permettendo in tal modo alle eventuali segnalazioni di FS, FI, N_1 e N_s di rivelarsi prima dell'effettuarsi della SCA.

e) Questo non avviene per la LN che occupando nell'esecuzione solo il canale esterno permette all'eventuale SCA che segue d'essere eseguita quando e' ancora in corso la lettura del nastro.
Si puo' pero' ovviare a questo inconveniente facendo precedere la SCA da una delle istruzioni che bloccano il proseguimento del programma in cui sono registrate, fino a terminata esecuzione dell'operazione di nastro in corso.
E' evidente che con una tale organizzazione si ottiene l'esecuzione della SCA proprio quando le possibili condizioni si siano verificate.

f) Qualora l'istruzione SCA rinviasse ad una istruzione diversa da salto verrebbero annullate tutte le segnalazioni memorizzate per conto dell'istruzione stessa.

g) L'esecuzione di uno dei salti e del relativo programmino annulla la segnalazione in questione.

h) Sfruttando la caratteristica g) e' conveniente far precedere ogni programma da una istruzione SCA seguita da istruzione diversa da salto per annullare eventuali segnalazioni rimaste memorizzate da precedenti elaborazioni.

Esempio :

SCA	ET_s	0016	T_m	F	0008
MA	LL	IIII	T_m	F	0016

SCHEMA RIASSUNTIVO	Istruzioni Nastri

Memoria - Nastri	:	LNa	LNi	RN	PRN	
Nastro - Memoria - Nastro	:	NDN	PIN	NAN		
Nastro - Nastro	:	TN				
Nastro	:	DUB	KN	AV		
Salti	:	SFS	SN_1	SN_s	SFI	SNO	SCA

Particolarita' dei Nastri

Informazioni - sono registrate e lette carattere per carattere e cioe' in serie.

Bit - ad ogni carattere corrispondono 8 bit disposti uno accanto all'altro nel senso della larghezza del nastro (6 bit definiscono un carattere, il 7° e' il bit di disparita' e l'8° il bit di temporizzazione).

Caratteri speciali dei nastri :
- α) il carattere " α " segnala l'inizio e la fine di un blocco;
- ?) il carattere " ? " inizia e termina il primo e l'ultimo blocco di una sequenza;
- Ø) il carattere " Ø " inizia e termina il primo e l'ultimo blocco di una informazione;
- Θ) il carattere " Θ " separa le informazioni costituenti un blocco organizzato per essere operato con la NDN.

Numero dei caratteri - registrabili in una bobina di nastro con blocchi di n caratteri in media, e' dato dalla seguente formula :

$$N = \frac{12.960.000 \cdot n}{n + 300}$$

Unita' nastro - collegabili al GUN sono al massimo 20. I caratteri che le distinguono sono : 123456789~ØABCDEFGHI

CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE							CARATTERE DI FUNZIONE
LNa	est. GUN	10	X	n_1	I	I	I	I	T_m	*
LNi	" "	10	X	n_1	I	I	I	I	T_m	K
RN	" "	10	X	n_s	I	I	I	I	T_m	M
PRN	" "	10	X	X	I	I	I	I	T_m	O
NDN	est. int. GUN		n_1	n_s	I	I	I	I	T_m	β
PIN	" " "	10	X	X	I	I	I	I	T_m	R
NAN	" " "	10	n_1	n_s	I	I	I	I	T_m	ε
TN	GUN		n_1	n_s	B	B	B	B	T_m	$
DUB	"	10	n	J	B	B	B	B	T_m	N
KN	est. GUN		X	n	I	I	I	I	T_m	P
AV	unita' nas.	10	X	n	X	X	X	X	T_m	-
SFS	interno	10 o 15	N	T_s	I	I	I	I	T_m	O
SN_1	"	10 o 15	P	T_s	I	I	I	I	T_m	O
SN_s	"	10 o 15	Q	T_s	I	I	I	I	T_m	O
SFI	"	10 o 15	ɀ	T_s	I	I	I	I	T_m	O
SNO	"	10 o 15	n	T_s	I	I	I	I	T_m	O
SCA	"	10 o 15	(	T_s	I	I	I	I	T_m	O

CAP. 9° : LA SIMULTANEITA' OPERATIVA DELL'ELEA 9003; LOGICA E UTILIZZAZIONE DEL 1° E 2° PROGRAMMA

9.1. Considerazioni generali

Alla memoria principale dell'ELEA 9003 si puo' accedere mediante due canali di trasferimento delle informazioni, uno interno ed uno esterno. Il canale interno collega la memoria principale agli organi interni della macchina; il canale esterno e' utilizzato per il collegamento con le unita' a nastro, tramite il relativo governo, e quando si opera direttamente fra due diverse zone di memoria.

L'istruzione dell'ELEA 9003 si svolge, come abbiamo gia' visto, in due fasi distinte : la fase preparatoria, durante la quale il canale interno della macchina e' impegnato in continuita' per la durata di nove periodi di cifra; la fase esecutiva, che varia a seconda della funzione che l'istruzione stessa deve svolgere, e che impegna sempre per un periodo di cifra il canale interno, prima che si inizi l'esecuzione vera e propria che potra' impegnare uno dei due canali.

Sotto questo punto di vista le istruzioni dell'ELEA 9003 sono raggruppate nel seguente modo :

1°) istruzioni che impegnano il canale interno;

2°) istruzioni che impegnano il canale esterno e il governo delle unita' a nastro magnetico;

3°) istruzioni che impegnano il canale interno ed il canale esterno;

4°) istruzioni che impegnano il canale interno, il canale esterno e il governo delle unita' a nastro magnetico;

5°) istruzioni che impegnano il governo delle unita' a nastro magnetico.

In particolare l'istruzione avvolgi nastro (AV) puo' essere eseguita simultaneamente con qualsiasi altra istruzione purche' quest'ultima non si riferisca all'unita' a nastro indicata nell'istruzione di avvolgimento.

In forma tabellare indichiamo le possibili sovrapposizioni tra le varie categorie di istruzioni precedentemente definite.

Gruppo di appartenenza dell'istruzione	Gruppi di appartenenza delle istruzioni che si possono svolgere in simultaneita' con la prima	
1°	2°	5°
2°	1°	-
3°	5°	-
4°	-	-
5°	1°	3°

Dall'esame di questa tabella risulta che i metodi di organizzazione di un programma sono due :

a) un programma che prevede l'utilizzazione di istruzioni doppie e quindi che esclude ogni simultaneita' operativa organizzata;

b) un programma che prevede l'utilizzazione delle sisultaneita' operative delle istruzioni dell'ELEA 9003.

Mentre nel caso a) lo svolgimento del programma si ripartisce linearmente sui due canali della macchina e quindi non vi e' alcuna ricerca di ottimizzazione dell'impegno degli stessi, in un programma di tipo b) questa ricerca e' necessaria ma, come vedremo, non puo' essere fatta dal programmatore.

Cio' significherebbe infatti tener conto dello svolgimento di ogni singola istruzione, predisporre il programma in modo da inserire tra due successive istruzioni esterne un numero opportuno di istruzioni interne, tale che la loro durata complessiva fosse uguale alla durata dell'istruzione esterna contemporaneamente eseguita.

La realizzazione della simultaneita' operativa si basa ed e' valida per le particolari caratteristiche logiche dell'ELEA 9003 che consentono alla macchina di operare su piu' sequenze di programma contemporaneamente.

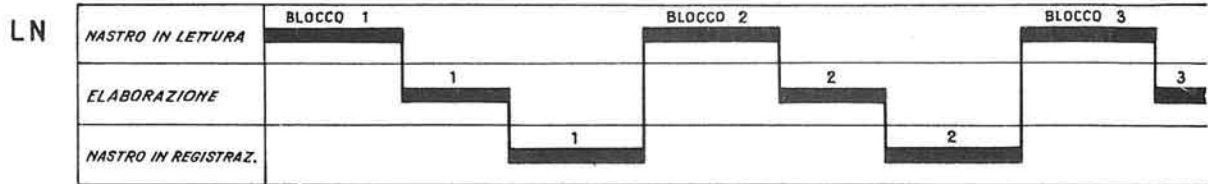
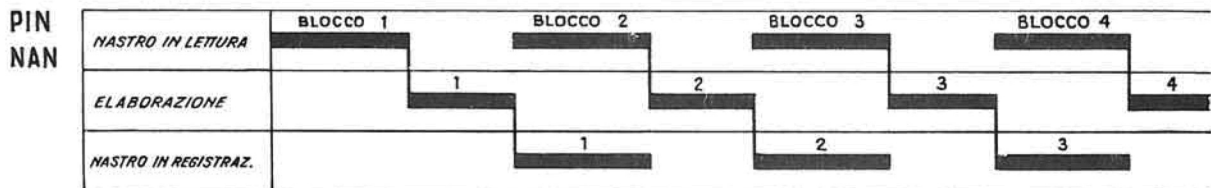
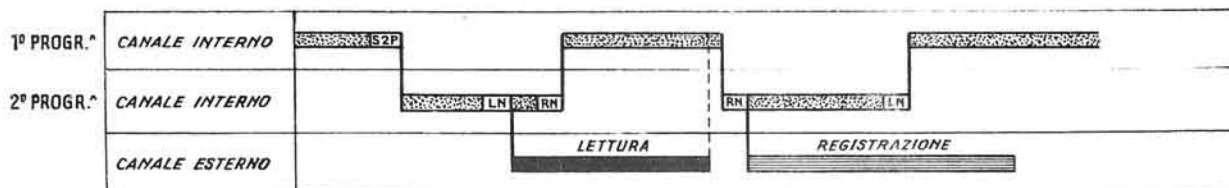
In questa analisi terremo conto delle sequenze definite di 1^{o} e di 2^{o} programma mentre esamineremo in altra sede la sequenza che e' propria del collegamento delle unita' in linea dell'ELEA 9003.

Esaminiamo ora le caratteristiche particolari dei due tipi di programmi precedentemente definiti.

9.2. Programma con istruzioni doppie

Non considerando la sequenza relativa al terzo programma, in questo tipo di organizzazione avremo lo svolgimento di una sequenza principale di programma : in essa ogni istruzione verra' eseguita in serie dalla macchina e in corrispondenza delle istruzioni doppie previste (PUM-MEM, NDN, PIN-NAN) avremo l'impegno contemporaneo dei due canali della macchina.

Una soluzione di questo tipo puo' per esempio essere adottata nei casi in cui i tempi di elaborazione siano brevi rispetto ai tempi di introduzione e di estrazione dei dati e percio' sia piu' conveniente sovrapporre questi ultimi.

NESSUNA SIMULTANEITA' OPERATIVA

SIMULTANEITA' IN LETTURA E REGISTRAZIONE NASTRI

SIMULTANEITA' DI ELABORAZIONE E LETTURA O REGISTRAZIONE

9.3. Programma con simultaneita' operative organizzate

Un programma di questo tipo, sempre non considerando la sequenza del terzo programma, comprendera' due sequenze di istruzioni : una definita di primo programma ed una di secondo programma.

Le caratteristiche principali di questa organizzazione sono due : a) la necessita' di una particolare organizzazione dei dati; b) i rapporti tra le due sequenze e la logica della simultaneita' interna.

9.3.1. L'organizzazione dei dati

Per utilizzare la possibilita' di sovrapposizione del tempo esecutivo dei due programmi e' necessario assegnare ad ogni flusso di informazioni piu' zone di memoria; in pratica sono sempre sufficienti due zone.

Si esamini infatti questo caso elementare. Dobbiamo introdurre da nastro magnetico delle informazioni su una zona di memoria che indichiamo con Zi (zona di introduzione), elaborarle depositando i risultati in un'altra zona di memoria che definiamo Ze (zona di estrazione), e infine estrarle da quest'ultima zona registrandole su nastro magnetico.

Utilizzando una sola zona per ogni flusso di dati, cioe' in pratica per ogni operazione esterna (nel caso nostro una LN e una RN), il programma risulterebbe cosi' strutturato :

1) LN sulla zona Zi.

2) Elaborazione dei dati contenuti in Zi e deposito dei risultati in Ze.

3) RN dalla zona Ze.

4) Di nuovo, LN nella zona Zi, ecc.

Ovviamente, non e' possibile attuare alcuna simultaneita', tra l'elaborazione e l'introduzione o la estrazione dei dati, perche' l'elaborazione sulla zona Zi non puo' aver luogo finche' non e' stata servita dalla LN e la estrazione dei risultati da Ze deve attendere che la fase di elaborazione sia terminata.

Se si ricorre ad una doppia zona di memoria per ogni operazione esterna, si potra' introdurre le informazioni una volta nella zona Zi e una volta nella zona Z'i, ed estrarle una volta dalla zona Ze e una volta dalla zona Z'e. Il programma risultera' cosi' strutturato:

1) LN sulla zona Zi

2) a) elaborazione dei dati contenuti in Zi e deposito dei risultati in Ze
   b) LN sulla zona Z'i

3) a) elaborazione dei dati contenuti in Z'i e deposito dei risultati in Z'e
   b) RN dalla zona Ze, seguita dalla LN sulla Zi

4) a) elaborazione dei dati contenuti in Zi e deposito dei risultati in Ze
   b) RN dalla zona Z'e seguita da LN sulla zona Z'i, ecc.

Le operazioni a e b risultano eseguite contemporaneamente, cosa che prima non era possibile.

Abbiamo detto che per avere simultaneita' operativa tra elaborazione ed introduzione o estrazione e' necessario servirsi di almeno due zone di memoria in alternanza : in effetti le zone utilizzate alternativamente potrebbero essere piu' di due con

qualche vantaggio di tempo in quanto, in alcuni casi, aumentando le zone utilizzabili si eliminano i tempi di attesa di uno dei due programmi. Ma si occuperebbero molte posizioni di memoria, non sempre disponibili e si appesantirebbe il programma con ulteriori istruzioni di servizio.

L'alternanza di zona per le operazioni esterne deve essere preparata dopo che e' stata eseguita l'operazione sulla zona interessata : ad esempio, dopo che si e' dato inizio all'istruzione LN sulla zona Zi, si predispone il relativo registro di modifica perche' la volta successiva la stessa LN sia eseguita sulla zona Z'i. In modo analogo si predispone l'alternanza per l'istruzione RN; in entrambi i casi si attua un meccanismo di flip-flop.

Generalmente le istruzioni di elaborazione interna saranno modificate da uno o piu' registri per poter operare dalle zone Zi a Ze e da Z'i a Z'e.

9.3.2. Rapporti tra le due sequenze di programma

L'esame delle due sequenze di programma e la contemporanea esecuzione vengono eseguiti dalla macchina secondo le seguenti regole logiche interne :

a) Se nel 1° programma sono in corso istruzioni che riguardano il canale interno e nel 2° istruzioni che riguardano il canale esterno o viceversa i due programmi agiscono simultaneamente.

b) Se nel 1° programma sono in corso istruzioni interne, e si verificasse che il 2° debba eseguire anch'esso un'istruzione interna, quest' ultimo prende la precedenza bloccando il 1° programma.

c) Se uno dei due programmi dovesse eseguire un' istruzione doppia (es. MEM, ÷ MM, ecc.), che impegni entrambi i canali, questo programma prende comunque la precedenza.

Come regola generale il secondo programma ha sempre la precedenza sul primo; in ogni caso pero' la precedenza viene presa dopo che sia stata eseguita la fase esecutiva dell'istruzione in corso.

Da quanto detto, per avere la massima sovrapposizione e' bene che sul 1° programma non vi siano mai istruzioni esterne, e che non si usino mai in nessun programma istruzioni doppie.

Inoltre e' bene nel 2° programma limitare allo stretto indispensabile l'uso di istruzioni interne.

E' consigliabile inoltre evitare l'uso degli stessi registri nei due programmi, mentre l' accumulatore dovra' essere usato su un solo programma.

In ogni modo se si volesse fare ugualmente uso dell'accumulatore e degli stessi registri e' necessario scaricarli prima e ricaricarli dopo averli usati.

9.4. Metodi di utilizzazione del 2° programma

Ogni metodo di utilizzazione del 1° e del 2° programma deve soddisfare alle due seguenti condizioni che costituiscono dei vincoli per il programmatore.

a) L'elaborazione sulla zona Z puo' essere eseguita solamente quando l'operazione esterna che serve quella zona ha avuto termine.

L'alternanza delle operazioni esterne deve quindi essere predisposta sul 2° programma e quella delle operazioni interne sul 1°.

b) Appena eseguita l'elaborazione interna e' necessario richiedere una operazione esterna sulla stes-

sa zona di memoria, per evitare eventuali tempi di attesa.

E' pertanto necessario programmare delle segnalazioni da parte del 2° programma al 1° programma ad ogni operazione esterna eseguita, e delle richieste di nuove operazioni esterne da parte del 1° al 2°.

Esistono due metodi di utilizzazione delle simultaneita' operative, realizzabili mediante le due istruzioni di salto speciale S2P e S2P*.

9.5. Descrizione del primo metodo

Al secondo programma si accede mediante l'istruzione S2P (salta al secondo programma) posta sul primo programma.

La lettura di questa istruzione provoca l'interruzione della sequenza in corso : infatti la macchina invia l'indirizzo di salto contenuto nella S2P nel registro indirizzi istruzioni del 2° programma e inizia l'esame delle istruzioni poste in questa seconda sequenza.

Da questo momento la macchina si comporta rispettando le regole di precedenza gia' descritte. Effettuato il salto cioe', i due programmi procedono parallelamente e il loro avanzamento contemporaneo dipende solamente dalle singole istruzioni poste sui due programmi. L'organizzazione del programma di lavoro deve prevedere l'esecuzione di una operazione interna di elaborazione e di una operazione esterna di introduzione e di estrazione alla volta.

Dato l'impegno dell'alternanza delle zone per le singole operazioni si deduce che se la operazione interna e' eseguita sulla zona Zi, contemporaneamente l'operazione esterna e' eseguita sulla zona Z'i.

E' chiaro allora che e' possibile avviare l'operazione interna sulla zona Z'i solo quando e' terminata la relativa operazione esterna.

In dettaglio, si avra' allora questo svolgimento : dopo un inizio di programma che provvede ad introdurre i primi dati nelle zone ed a richiedere l'introduzione dei dati successivi nelle zone alterne ad essi destinate, viene avviato il secondo programma che esegue le operazioni esterne richieste. Nel frattempo il 1° programma provvede ad eseguire le necessarie elaborazioni e, posizionando opportuni deviatori, ad annotare le operazioni esterne che piu' tardi dovra' richiedere al 2° programma.

Al termine delle elaborazioni, il 1° programma passa ad una istruzione di salto S2P che conduce ad uno STOP. Ora, l'istruzione S2P ha la proprieta' di non venire eseguita fino a che e' in corso il 2° programma : cio' obbliga il 1° programma ad attendere il termine delle operazioni esterne in corso, o, se queste sono gia' terminate, gli permette di passare oltre dopo un'attesa di soli 25 periodi di cifra (15 per la istruzione S2P, 10 per lo STOP).

Solamente dopo aver eseguita l'istruzione S2P il 1° programma provvede a richiedere al 2° le operazioni esterne precedentemente annotate.

Soddisfatte tali richieste il 1° programma avvia nuovamente il 2° con una istruzione S2P e riprende la elaborazione.

Tale organizzazione da' la sicurezza che le due condizioni richieste siano soddisfatte : infatti, appe

1° PROGRAMMA
2° PROGRAMMA
Esegue 1^ operaz. esterna
Prepara alternanza zona
S 2 P
E' terminata l'elaborazione?
SI
STOP
NO
Operazione interna
Finita zona dell'operazione interna ?
NO
SI
Alterna zona
Prepara la richiesta di un operazione esterna su quella zona
S 2 P
Completa le richieste di operazioni esterne
C'è da eseguire una data operazione esterna ?
NO
SI
Esegue operazione esterna
Alterna zona
Ci sono da eseguire operazioni esterne d'altro tipo ?
NO
SI
Esegue operazioni esterne
Alterna zona
STOP
1° METODO

1º PROGRAMMA
2º PROGRAMMA
1 → J
J → Zi
J + 1 → J
·β1 ·γ2
·δ3 ·μ1
S2P
STOP
γ
Elaboraz.
Zi → Ze
Elaboraz.
Z'i → Z'e
·γ1 ·τ1
·γ2 ·τ2
S2P
μ
·γ3
τ
·β1 · δ1
·β2 ·δ2
SI
J > n?
NO
·β3 μ2
δ
Z'e → reg.
Ze → reg.
β
J → Z' i
J → Zi
J + 1 → J
STOP

FIGURA 1

na terminata l'elaborazione sulla zona Z, il 1° programma ha provveduto, ad esempio in un ciclo n, a richiedere l'operazione esterna che serve quella zona.

Durante il ciclo n + 1, viene effettuata sia l'elaborazione sulla zona alterna, che l'operazione esterna sulla zona Z : all'inizio del ciclo n + 2, in cui si dovranno elaborare le informazioni su questa ultima zona, sono terminate tutte le operazioni esterne richieste durante il ciclo n, e quindi certamente e' terminata anche l'operazione sulla zona Z.

Caratteristica di questo metodo e' quindi di assicurare una protezione globale contro il pericolo di iniziare l'elaborazione su una zona non ancora servita, imponendo al 1° programma di attendere anche quando le operazioni esterne non ancora effettuate non riguardano l'elaborazione che dovrebbe iniziare.

Per tale motivo, tale metodo non si adatterebbe, per esempio, al caso in cui fossero previste piu' di due zone di alternanza per ogni operazione esterna.

In conclusione questo metodo implica che il programma attenda la fine del 2° programma in corso prima di riprendere il ciclo delle operazioni interne, e il successivo rilancio del 2° programma mediante una S2P posta sul 1°.

Lo schema logico e il diagramma a blocchi della figura 1 definiscono in tutti gli aspetti l'applicazione di questo primo metodo. Con J verra' indicato il generico blocco da introdurre in memoria; con n il numero dei blocchi da elaborare e con β, γ, δ, μ, τ i deviatori utilizzati.

9.6. Descrizione del secondo metodo

Questo metodo si basa non su una condizione rigida di rapporti tra il 1^{o} ed il 2^{o} programma, come avveniva nel primo metodo, ma su scambi di informazioni programmati tra le due sequenze.

Gli scambi da realizzare tra i due programmi sono :

a) Il 1^{o} programma "segnala", col massimo anticipo, le proprie future necessita', perche' le zone interessate siano rifornite di dati o ne siano sgomberate. Chiameremo tali comunicazioni "segnalazioni".

b) Il 1^{o} programma "richiede" l'esecuzione di una data operazione esterna su una zona; se la "richiesta" non puo' essere subito soddisfatta, il 1^{o} programma attende.
Dato che il 2^{o} programma, quando ha ricevuto la "segnalazione" fara' il possibile per prepararsi a soddisfare la "richiesta" che seguira', senza bisogno di solleciti, sara' opportuno effettuare la richiesta all'ultimo momento.
Fisicamente, sul programma, non e' necessario che ci sia una corrispondenza 1-1 tra "segnalazioni" e "richieste", perche', mentre le segnalazioni servono ad avvisare il 2^{o} programma della necessita' di una operazione esterna, le richieste servono ad assicurare il 1^{o} programma che l'operazione esterna su una determinata zona e' stata eseguita.

Sarebbe necessario introdurre un ulteriore deviatore che consentisse di accedere alla istruzione solo quando il 2^{o} programma ha raggiunto lo STOP e di scavalcarla in via normale, quando il 2^{o} programma e' in corso.

Si puo' concludere affermando che il 2^{o} metodo, a differenza del 1^{o}, assicura in tutti i casi la mas-

1° PROGRAMMA

2° METODO

1º PROGRAMMA
2º PROGRAMMA
·β2 ·α1 ·γ1
·π1
j = i = i' = e = e' = 1
1 → K
0 → h
J → Mi
S2P *
β
1
2
i < i'
SI
α
1
2
oscillatore
STOP
NO
Elaboraz.
Mi' → Me
h+1 → h
i' + 1 → i'
e + 1 → e
h = m ?
SI
NO
·β2
·β1
γ
1
2
·γ2
·γ1
k = m ?
SI
NO
j + 1 → j
i + 1 → i
j → Mi
k + 1 → k
j = n ?
NO
SI
·α2
·π2
h = 0 ?
SI
STOP
NO
Me' → reg.
k - 1 → k
h - 1 → h
e' + 1 → e'
π
2
1

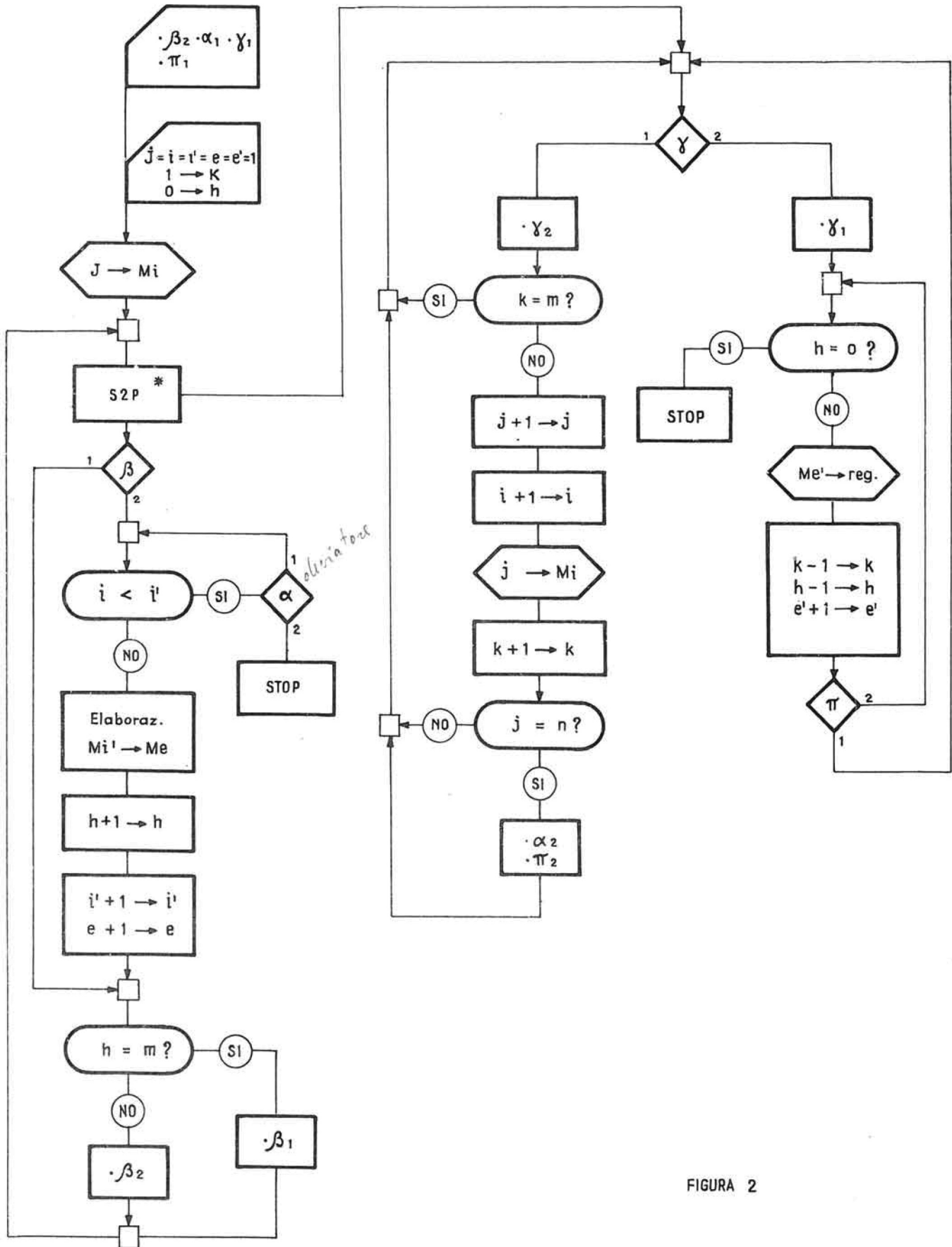



FIGURA 2

sima sovrapponibilita' logicamente e tecnicamente possibile.

Basti pensare ad un programma con due sole operazio ni esterne, una LN e una RN, in cui per ogni blocco letto vi siano piu' blocchi da registrare, per esem pio tre in media.
Il 2° metodo ci permette di accedere alle elabora- zioni dei blocchi successivi anche quando non e' ter minata la serie delle registrazioni su nastro dopo l'elaborazione di ogni blocco, con vantaggio tanto piu' notevole quanto piu' varia puo' essere la dura ta dell'elaborazione dei singoli blocchi.

Lo schema logico ed il diagramma a blocchi della fi gura 2 definiscono l'applicazione di questo secon- do metodo.

In particolare nella fig. 2 abbiamo indicato con :

J = blocco generico introdotto in memoria

n = numero dei blocchi da elaborare

m = numero delle zone di memoria disponibili

i = indice della zona di memoria in cui vengono in trodotti i dati da elaborare

i' = indice della zona di memoria in elaborazione

e = indice della zona di memoria contenente dati elaborati

e' = indice della zona di memoria da cui vengono e stratti i dati elaborati

k = numero delle zone di memoria gia' impegnate

h = numero delle zone di memoria con dati pronti per essere estratti.

Evidentemente le zone di memoria coincidono e sono state indicate con i diversi quattro indici (i, i',

e, e') secondo la fase di programma in cui si trovano.

Si intende inoltre che le zone di memoria sono disposte circolarmente, cioe' all'ultima zona, i = m, seguira' ancora la prima zona, i = m + 1 = 1 .

Le richieste di cui precedentemente abbiamo detto, sono configurate con la domanda (i < i' ?) cioe'quando ci si chiede se la zona che dobbiamo elaborare sia gia' stata fornita di dati.

Le segnalazioni sono invece configurate con il contatore "h" che segnala al 2º programma se vi sono delle zone con dati elaborati da estrarre ed in seguito da rifornire con nuovi dati.

In conclusione, questo metodo implica che il 2º programma puo' essere avviato una sola volta e che il 1º programma attende, prima di elaborare, che siano stati introdotti ed estratti i dati nelle e dalle zone interessate.

E' pero' possibile che ad un certo punto, il 2º programma arrivi allo STOP mentre il 1º sta ancora eseguendo operazioni interne.

In questo caso, terminata l'operazione interna, il 1º programma riavviera' il 2º con l'istruzione S2P*.

La caratteristica operativa dell' S2P*, consente di inserirla nel ciclo principale del 1º programma. Il governo dell'unita' centrale la leggera' nel corso di ogni ciclo; se la sequenza di 2º programma sara' in corso, non blocchera' lo svolgimento del 1º, ma ne ritardera' di soli 15 periodi di cifra l' esecuzione.

In caso contrario, se nel corso del lavoro il 2º programma avra' raggiunto lo STOP, il salto S2P* ver-

ra' eseguito e dara' corso ad una nuova sequenza del 2^o programma.

Si puo' notare che il ricorso alla S2P per il secondo metodo di utilizzazione richiederebbe un dispositivo di sicurezza atto a neutralizzare la sua funzione bloccante della sequenza di 1^o programma in cui e' inserita.

Salta al 2° Programma		S2P
interno	(101110) α T_s I I I I T_m 0	10 o 15

α : carattere di eventualita'

T_s : registro che ricorda l'indirizzo della posizione p8 dell'istruzione

I I I I : indirizzo dell'istruzione posta nella sequenza di 2° programma alla quale si deve saltare

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro T_s

0 : salta se si e' verificata l'eventualita' α.

a) L'istruzione posta nel 1° programma comanda l'inizio di una sequenza di 2° programma.

b) Se e' gia' in corso una sequenza di 2° programma la S2P viene riciclata fino al termine dell'esecuzione del 2° programma; eseguito lo STOP di 2° programma la S2P riavvia il programma stesso.

Salta al 2° Programma *		S2P*
interno	(010010) δ T_s I I I I T_m 0	15

δ : carattere di eventualita'

T_s : registro che ricorda l'indirizzo della posizione p8 dell'istruzione

I I I I : indirizzo dell'istruzione posta nella sequenza di 2° programma alla quale si deve saltare

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro T_s

0 : salta se si e' verificata l'eventualita' δ

a) L'istruzione posta nel 1° programma comanda l'inizio di una sequenza di 2° programma.

b) Se e' gia' in corso una sequenza di 2° programma la presente istruzione non blocca il programma su cui e' posta.

SCHEMA RIASSUNTIVO — Istruzioni di salto al 2° programma

Istruzioni : S2P S2P*

Particolarita' sulle istruzioni di 2° programma

Entrambe le istruzioni : 1°) permettono l'avvio di una sequenza di 2° programma;

2°) possono essere poste solo nel 1° programma.

S2P : Se e' gia' in corso una sequenza di 2° programma l'istruzione viene riciclata fino a che il 2° programma non sia terminato.

Eseguito lo STOP di 2° programma la S2P puo' quindi riavviarlo.

S2P* : Se e' gia' in corso una sequenza di 2° programma la presente istruzione non blocca il programma su cui e' posta.

CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE	CARATTERE DI FUNZIONE
S2P	interno	10 o 15	α T_s I I I I T_m	0
S2P*	"	15	δ T_s I I I I T_m	0

CAP. 10° : LA TERZA SEQUENZA DI PROGRAMMA

10. 1. Generalita'

1) All'unita' centrale del sistema Elea 9003 possono essere collegate unita' in linea capaci di funzionare simultaneamente e di operare senza interrompere lo svolgimento del 1° e del 2° programma.

2) Le unita' collegabili sono :
   - lettori di schede
   - perforatori di schede
   - lettori di nastro perforato
   - lettori e perforatori di schede
   - stampanti

   Le varie unita' numerate da 1 a 10 vengono distinte in fase di elaborazione dal proprio numero indicativo.

   Nell'uso normale troviamo simultaneamente collegate unita' di diverso tipo anche se teoricamente esiste la possibilita' di porre in linea dieci unita' con medesima funzione.

3) Ciascuna di queste unita' e' diretta da un proprio governo che contiene l'apparecchiatura logica necessaria al suo funzionamento e una memoria a nuclei sufficiente a contenere i caratteri necessari per ogni ciclo meccanico di lavoro :

   una riga stampa,
   una scheda,
   un blocco di nastro perforato (max. 104 caratteri)

4) L'alternarsi dell'entrata e dell'uscita dei caratteri dalle varie memorie, la distribuzione dei dati alle diverse unita' collegate e la logica che le regola, sono affidate ad un sincronizzato

re che armonizza tutte le operazioni di terzo programma.

10.2. Caratteristiche logiche del 3° programma

1) L'ordine di lavoro alle singole unita' viene trasmesso dall'unita' centrale, che procede quindi in parallelo con esse, senza che queste ultime rallentino ne' intralcino le operazioni interne; il sincronizzatore provvede a smistare gli ordini e le informazioni alle diverse apparecchiature; ogni unita' chiamata ad operare esegue il lavoro secondo le peculiari modalita' ed il proprio tempo meccanico.

   Le memorie di transito delle singole unita' di estrazione vengono fornite delle informazioni in uscita dall'unita' centrale che provvede pure a prelevare le informazioni in entrata dalle unita' di introduzione.

   Non e' necessaria ad ogni trasferimento da o a unita' centrale un'operazione di azzeramento delle memorie di transito in quanto queste vengono azzerate automaticamente.

2) Durante il collegamento con il calcolatore, le unita' di introduzione ed estrazione assumono tre diversi stati possibili; e' :

   "inattiva" l'unita' che pur collegata con l'unita' centrale attraverso il proprio governo, non sia avviata;

   "avviata" l'unita' che sia stata chiamata ad operare almeno una volta mediante un'istruzione. Lo stato di "avviata" puo' durare per molti cicli meccanici di lavoro;

"occupata" l'unita' che oltre ad essere avviata sia stata effettivamente messa in funzione da una istruzione esecutiva.

Ogni unita' avviata ed occupata passa automaticamente allo stato di avviata-non occupata al termine di ogni ciclo meccanico di lavoro; passa allo stato di inattiva mediante un'apposita istruzione di STOP.

La simultaneita' di esecuzione delle operazioni interne della macchina e del lavoro delle unita' collegate e' ottenuto mediante una terza sequenza di istruzioni eseguibili dall'elaboratore parallelamente alle sequenze di 1° e 2° programma.

L'uso delle diverse sequenze non comporta intralci ne' inconvenienti alla corretta e coerente esecuzione delle istruzioni dei singoli programmi.

Esistono prestabilite priorita' sul canale interno, tre diversi registri indirizzo istruzioni, e dispositivi distinti per ogni programma per la conservazione separata delle seguenti segnalazioni circa determinate eventualita' verificabili nel corso dell'elaborazione: risultato dei confronti ($=$, $\neq$, $>$, $<$) e overflow nei registri e nella memoria.

Si deve inoltre osservare che contrariamente alle istruzioni di canale esterno che vengono poste preferibilmente sul 2° programma, per sole ragioni di logica, le istruzioni di comando delle apparecchiature in linea devono essere raccolte obbligatoriamente sul 3° programma per ragioni funzionali. Questa disposizione risulta inoltre logica specie se si considerano le norme che stabiliscono le priorita' sui canali.

La priorita' sul canale interno e' riservata :

prima al 3° programma che per l'esecuzione delle istruzioni che lo riguardano non richiede il canale interno se non per periodi di tempo assai limitati;
quindi al 2° programma che si svolge nella massima parte con impegno del solo canale esterno;
infine al 1° programma che impegna il canale interno sia in fase preparatoria che in fase esecutiva.

E' evidente che dette priorita' sono studiate in modo da permettere ad ogni programma la massima continuita' senza che l'esecuzione degli altri programmi subisca per questo eccessivi tempi di attesa.

Risulta pertanto logico che dovendo i tre programmi ricorrere al canale interno per il loro svolgimento, su questo canale debba aver la precedenza il programma che meno lo impegna.

In pratica il 3° programma cede la precedenza sul canale interno :

1° ogni qualvolta le memorie di transito dei governi delle unita' in linea risultino fornite dei caratteri necessari ad un ciclo meccanico di lavoro;

2° ogni qualvolta si verifichi un impedimento all'esecuzione di una delle istruzioni registrate in detto programma.

La precedenza al terzo programma invece non viene immediatamente ceduta in due particolari casi :

1° allorquando sia stata letta su 1° o 2° programma una istruzione preparatoria tipo PUM, PRN, PIN . Nel qual caso il canale interno viene ceduto solo

dopo l'esecuzione dell'istruzione esecutiva tipo MEM, + MM, - MM, TN, NAN;

2° allorquando sia stata letta ed eseguita su 1° o 2° programma una istruzione di salto. Nel qual caso il canale viene ceduto solo dopo l'esecuzione dell'istruzione a cui il salto rinvia.

La terza sequenza di programma :

a) puo' essere avviata da una istruzione particolare di salto S3P, registrata indifferentemente nel 1° o nel 2° programma;

b) puo' essere frazionata in piu' sequenze singolarmente avviabili ma non contemporaneamente eseguibili in quanto l'elaboratore puo' essere impegnato nell'esecuzione di una sola sequenza di 3° programma;

c) puo' eseguire istruzioni di qualsiasi genere pur avendo come funzione specifica i comandi relativi alle apparecchiature in linea.

10.3. Fasi di svolgimento del 3° programma

1) Una istruzione S3P da inizio all'esecuzione della sequenza. Appositi registri, distinti da quelli con ugual funzione, di 1° e di 2° programma, conservano gli indirizzi e le segnalazioni relative alle istruzioni di 3° programma.

2) Vengono quindi percorse le istruzioni della sequenza secondo le normali modalita' di funzionamento, ma con i seguenti effetti :

   a) l'istruzione SUO avvia l'unita' interessata;

   b) l'istruzione TOL esegue un ciclo completo di lavoro : trasferimento dei caratteri alla memoria di transito ed elaborazione relativa;

c) le altre istruzioni compiono la loro normale funzione.

3) Ritorno alla 1° istruzione SUO eseguita, nelle seguenti possibili condizioni :

a) l'unita' relativa alla SUO non e' occupata; nel qual caso il salto non viene effettuato e si eseguono l'istruzione TOL e successive;

b) l'unita' relativa alla SUO e' occupata ma almeno una delle altre unita' si e' resa disponibile; nel qual caso viene effettuato il salto all'istruzione indicata dalla SUO e la sequenza viene percorsa fino alla prossima SUO che si trovi nella condizione c);

c) l'unita' relativa alla SUO e' occupata e cosi' pure tutte le altre unita'; nel qual caso il 3° programma si arresta su detta istruzione.

4) Esecuzione infine degli STOP relativi alle unita' in linea, con i seguenti possibili effetti :

a) se lo STOP viene eseguito quando ancora non e' letto almeno uno degli altri STOP esistenti nella sequenza di 3° programma, l'unita' interessata passa allo stato di "inattiva" senza che lo svolgimento del programma venga interrotto;

b) se lo STOP e' eseguito dopo l'avvenuta lettura degli altri STOP l'unita' passa allo stato di inattiva ed il programma si arresta.

10.4. Funzione e logica delle istruzioni S3P, SUO, TOL, STOP

L'istruzione S3P a) comanda l'inizio di una sequenza di 3° programma;

b) attiva tutte le unita' che vengono percorse dal 1° ciclo di 3° programma;

c) viene riciclata qualora sia letta mentre e' gia' in corso una sequenza di 3° programma;

d) puo' riavviare una sequenza di 3° programma non appena ne sia terminata una precedente.

L'istruzione SUO permette

a) di avviare l'unita' a cui e' riferita;

b) l'arresto del programma nel caso che nessuna unita' sia disponibile;

c) il riavvio del programma ad ogni segnalazione di unita' disponibile.

- Dopo l'arresto su una SUO, il riavvio del programma avviene con la lettura dell' istruzione stessa.

- Nel caso la SUO richiamasse una unita' non collegata, il calcolatore non esegue la sequenza di programma relativa a tale unita'.

L'istruzione TOL deve seguire immediatamente nel programma la SUO relativa.

Essa permette :

a) di eseguire l'effettivo trasferimento dei dati fra la memoria dell'unita' centrale e le memorie di transito delle unita' in linea;

b) di predisporre l'unita' in linea all'esecuzione del ciclo operativo successivo : autorizzazione alla prossima lettura, perforazione o stampa, preceduta o seguita dall'azzeramento della memoria di transito.

L'istruzione STOP appare generalmente al termine di ogni sequenza d'istruzioni relative ad una unita' in linea. Essa permette :

a) di riportare una unita' allo stato di inattiva non appena sia stata completata l'operazione per la quale era stata chiamata;

b) di arrestare il 3° programma una volta eseguiti tutti gli STOP delle unita' avviate.

Va rilevato che l'istruzione STOP non arresta il ciclo operativo in corso, ma le operazioni atte a predisporre l'unita' in linea al successivo ciclo di lavoro dopo l'azzeramento automatico della memoria di transito.

10.5. L'istruzione S3P*

Mediante questa istruzione si vuole ovviare all'inconveniente generato dall'uso della sola istruzione S3P normale, che non permette di realizzare un meccanismo dinamico di sovrapposizione tra 1° o 2° programma e 3° programma, basato su uno scambio di richieste e di segnalazioni.

Caratteristica logica dell'istruzione S3P* e' infatti quella di rendere possibile un ciclo intero di 3°

programma ogni qualvolta se ne presenti la necessita' durante lo svolgimento dei primi due programmi, permettendo in tal modo di avviare una sequenza relativa ad una unita' in linea temporaneamente inutilizzata.

Differenza rilevante tra l'istruzione S3P e l'istruzione S3P* e' che mentre la prima attiva le unita' in linea, la S3P* puo' soltanto avviarle.

Ne consegue che :

1° tutte le unita' interessate dalla sequenza di 3° programma devono essere attivate dall'istruzione S3P e conseguentemente percorse durante il 1° ciclo di 3° programma;

2° una unita' resasi inattiva non puo' essere riattivata se non dopo che tutte le unita' considerate nel 3° programma siano tornate allo stato di inattive.

Si osservi inoltre che l'uso della istruzione S3P* richiede l'impiego nel 3° programma di una istruzione SUO relativa ad una unita' inesistente (SUO su unita' fittizia).

L' istruzione SUO su unita' fittizia deve apparire in testa alla sequenza di istruzioni ed essere letta nel 1° ciclo di 3° programma.

10.6. L'organizzazione di un 3° programma che preveda l'uso della istruzione S3P* dovra' pertanto essere legata alle seguenti norme.

1) Sul 1° o 2° programma deve essere registrata una istruzione S3P che permetta l'attivazione di tutte le unita' interessate all'elaborazione.

2) La S3P deve rinviare direttamente o indirettamente all'istruzione SUO su unita' fittizia che si trova in testa al 3^{o} programma. Solo in questo modo l'unita' puo' risultare facente parte del 3^{o} programma.

3) Le sequenze relative ad ognuna delle unita' in linea interessate devono essere precedute da un deviatore che resta aperto durante il primo ciclo di 3^{o} programma.

4) Se una unita' non e' da utilizzarsi immediatamente, il deviatore che ne introduce la sequenza relativa deve venir chiuso, cioe' indirizzato al successivo deviatore, mediante una istruzione apposita che lo deve seguire nel programma.

5) Successivi posizionamenti dei deviatori possono essere disposti da istruzioni registrate indifferentemente in uno dei tre programmi a seconda delle necessita' di programmazione.

6) L'istruzione S3P* non deve rinviare alla SUO su unita' fittizia ma all'istruzione che la segue immediatamente perche' la lettura della SUO provocherebbe la disponibilita' del canale interno per il 1^{o} o 2^{o} programma.

Ripresa automatica del canale interno da parte del 1^{o}, 2^{o} e 3^{o} programma.

- Il 1^{o} o 2^{o} programma riprendono la precedenza se si verifica nel 3^{o} una delle seguenti eventualita' :

  1^{o} tutte le unita' in linea sono allo stato di "occupata" e il 3^{o} programma e' fermo sull'istruzione SUO in corrispondenza della quale tale eventualita' si e' verificata (programma che prevede il solo uso dell'istruzione S3P);

2^o tutte le unita' in linea sono allo stato di "occupata" ed il 3^o programma e' fermo sull'istruzione SUO su unita' fittizia(programma che prevede l'uso dell'istruzione S3P*).

- Il 3^o programma riprende la precedenza se e' presente almeno una delle seguenti condizioni :

  1^o una unita' precedentemente occupata si e' resa disponibile;

  2^o richiesta di ciclo di 3^o programma da parte del 1^o o 2^o programma mediante l'istruzione S3P*;

  Nell'organizzazione che prevede l'uso della S3P* nonostante l'arresto sulla SUO su unita' fittizia, nel riavvio di un ciclo di 3^o, detta istruzione non e' considerata e viene invece letta immediatamente la istruzione che segue;

  con l'uso della sola S3P l'istruzione SUO da cui il programma riparte viene viceversa letta ed eventualmente eseguita.

Automatismo relativo ai segnali di unita' disponibile

Il 3^o programma, indipendentemente dall'uso della S3P*, viene percorso fintanto che perdura la segnalazione di unita' disponibile.

Questa segnalazione ha tuttavia caratteristiche diverse a seconda dell'organizzazione scelta.

Infatti coll'uso dell'istruzione SUO su unita' fittizia, eventuali segnalazioni di unita' disponibile permangono solo temporaneamente, di modo che se per una ragione qualsiasi (chiusura di un deviatore) l'unita' disponibile non viene occupata durante il ciclo richiesto, tale segnalazione si perde, e solo un nuovo comando originato da una S3P* o da una nuova segnalazione di disponibilita' puo' permettere un ciclo di lavoro all'unita' resasi precedentemente disponibile.

Salta al programma esterno		S3P
interno	(011101) , T_s I I I I T_m 0	10 o 15

, : carattere di eventualita'

T_s : registro che ricorda l'indirizzo della posizione p8 dell'istruzione

I I I I : indirizzo dell'istruzione posta nella sequenza di programma esterno alla quale si deve saltare

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome del registro T_s

0 : salta se si e' verificata l'eventualita' ,

a) L'istruzione posta nel 1º e 2º programma comanda l'inizio di una sequenza di programma esterno.

b) Se e' gia' in corso una sequenza di programma esterno :

   1º) la presente istruzione viene riciclata finche' la sequenza in corso non e' terminata;

   2º) terminata l'esecuzione della sequenza di programma esterno, la S3P puo' riavviarla.

Salta al programma esterno *		S3P*
interno	(010101) = T_s I I I I T_m 0	10

= : carattere di eventualita'

T_s : registro che ricorda l'indirizzo della posizione p8 dell'istruzione

I I I I : indirizzo dell'istruzione che segue la SUO su unita' fittizia

T_m : il registro T_m puo' modificare l'indirizzo IIII e il nome del registro T_s

0 : salta se si e' verificata l'eventualita' =

a) La presente istruzione suppone avviata una sequenza di 3º programma mediante una precedente S3P normale.

b) Per essere utilizzata e' necessario che la sequenza di 3º programma abbia una SUO su unita' fittizia.

c) Se e' gia' in corso una sequenza di programma esterno la S3P* non blocca il programma su cui e' posta.
La S3P* viene sempre eseguita anche se e' in corso una sequenza di 3º programma, non blocca mai il programma in cui e' posta; permette un ciclo intero di 3º programma ogni qualvolta ne sorga la necessita'.

d) Il ciclo di 3º programma ottenuto mediante S3P* parte dall'istruzione successiva alla SUO su unita' fittizia e si arresta invece su quest'ultima.

Salta se l'unita' U e' occupata		SUO
interno	(011010) % U I I I I T_m 0	10 o 15

% : carattere di eventualita'

U : nome dell'unita' in linea

I I I I : indirizzo dell'istruzione a cui si deve saltare

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che il nome dell'unita' U

0 : salta se si e' verificata l'eventualita' % .

a) In ogni sequenza di programma esterno compaiono tante SUO quante sono le unita' in linea interessate.

b) La prima volta che la sequenza viene percorsa nessuna SUO viene eseguita, ma l'unita' in linea U, specificata in p7, viene avviata.

c) La presente istruzione serve pure a specificare l'unita' interessata dall' istruzione TOL che segue.

Trasferisci On Line		TOL
interno	L L I I I I T_m J (110001)	10 ÷ 1

L L : lunghezza dell'operando

I I I I : indirizzo di memoria dell'operando da trasferirsi

T_m : il registro T_m puo' modificare sia l'indirizzo IIII che la cifra delle unita' della lunghezza LL

J : trasferisce dalla memoria principale alla memoria di transito oppure viceversa, a partire dall'indirizzo IIII per lunghezza LL e per indirizzi crescenti.

a) L'istruzione trasferisce da oppure a memoria di transito secondo che l'unita' in linea specificata dalla SUO immediatamente precedente sia un lettore di schede o un lettore fotoelettrico di banda perforata, oppure una stampante o un perforatore di schede.

Salta se errore in unita' in linea			SEL
interno, unita' in linea U	(011110) γ U I I I I T_m 0		10 o 15

a) Salta se si e' verificato un errore nell'unita' in linea U, specificata in p7.

b) La segnalazione di errore su unita' in linea presenta diverse caratteristiche a secondo del genere di apparecchiature nel quale l'errore si verifica; infatti la segnalazione avviene :

<u>per il lettore di schede</u> : al termine della lettura della scheda successiva a quella in cui l'errore si e' verificato

<u>per il perforatore di schede</u> : al termine della perforazione della 2° scheda successiva a quella in cui l'errore si e' verificato

<u>per la stampante</u> : terminato un ciclo di stampa (prima della TOL successiva)

Arresto di unita' in linea			STOP
interno	(111001) 0 U X X X X # 0		10

0	: carattere di eventualita'
U	: unita' in linea
X X X X	: posizioni non utilizzate
#	: posizione non utilizzata
0	: arresta l'unita' in linea U, specificata in p7.

a) L'unita' in linea U passa allo stato "inattiva"

b) Se una o piu' unita' in linea passano allo stato "inattiva" la sequenza di programma esterno continua a svolgersi per le rimanenti.

c) Solo quando tutte le unita' in linea hanno ricevuto la segnalazione di Stop, il 3° programma si arresta.

d) L'arresto del 3° programma non blocca gli altri programmi eventualmente in corso.

e) Lo Stop non arresta immediatamente l'unita' interessata ma solo al termine dell'ultimo ciclo meccanico di lavoro.

SCHEMA RIASSUNTIVO	Istruzioni di 3° programma

Istruzioni : S3P S3P*

SUO, TOL, SEL, STOP

Particolarita' del 3° programma	CODICE SIMBOLICO	CANALI IMPEGNATI	TEMPO IN PERIODI DI CIFRA	CONFIGURAZIONE	CARATTERE DI FUNZIONE
S3P : non puo' essere eseguita se e' in corso il 3° programma, e viene riciclata con conseguente blocco del programma in cui e' registrata.	S3P	interno	10 o 15	, T_s I I I I T_m	
S3P* : permette un ciclo di 3° programma ogni qualvolta viene letta, anche se detto programma e' gia' in corso.	S3P*	"	10	= T_s I I I I T_m	0
SUO, TOL, SEL, STOP : per le particolarita' riguardanti queste istruzioni si rimanda agli schemi relativi.	SUO	"	10 o 15	% U I I I I T_m	0
	TOL	inter. e Unita' in linea	10 ÷ 1	L L I I I I T_m	J
	SEL	interno	10 o 15	γ U I I I I T_m	0
	STOP	"	10	0 U X X X X #	0

Unita' collegabili : lettori di schede
perforatori di schede
lettori di nastro perforato
lettori e perforatori di schede
stampanti.

N. unita' collegabili: 10, omogenee o di tipo diverso.

Stati possibili di una unita' collegata :

inattiva = non avviata
avviata = avviata ma non occupata
occupata = avviata e occupata

Priorita' del 3° sul 1° e 2° programma : ogni qualvolta sia disponibile una unita' eccettuati i casi seguenti :

a) nel 1° o nel 2° sia letta la prima di una istruzione doppia : PUM, PIN, PRN ;
b) nel 1° o nel 2° sia letta ed eseguita una istruzione di salto qualsiasi.

10.7. Esempio di 3° sequenza con uso della sola S3P

Generalita'

- Nell'esempio seguente si fa riferimento ad una sequenza di 3° programma avviata dal 1° mediante S3P.
- La sequenza e' costituita da 2 cicli fondamentali riferentisi rispettivamente a 2 unita'. Nel caso specifico, unita' 1, 2.
  L'unita' 1 e' supposta essere un perforatore di schede, l'unita' 2 un lettore di schede.
- Sono supposte gia' preparate le zone di memoria che saranno operate nelle istruzioni TOL (una per ognuno dei sopraddetti cicli).
- I registri T utilizzati sono T_1, T_2 (rispettivamente associati alle unita' 1, 2), che servono sia per la conta del numero di schede, sia per la modifica dell'indirizzo della TOL per passare da una zona predisposta per una scheda, alla successiva.
- Schede da perforare (unita' 1) : 6 su 80 colonne= 480 caratteri.
  Schede da leggere (unita' 2) : 20 su 40 colonne= 800 caratteri.

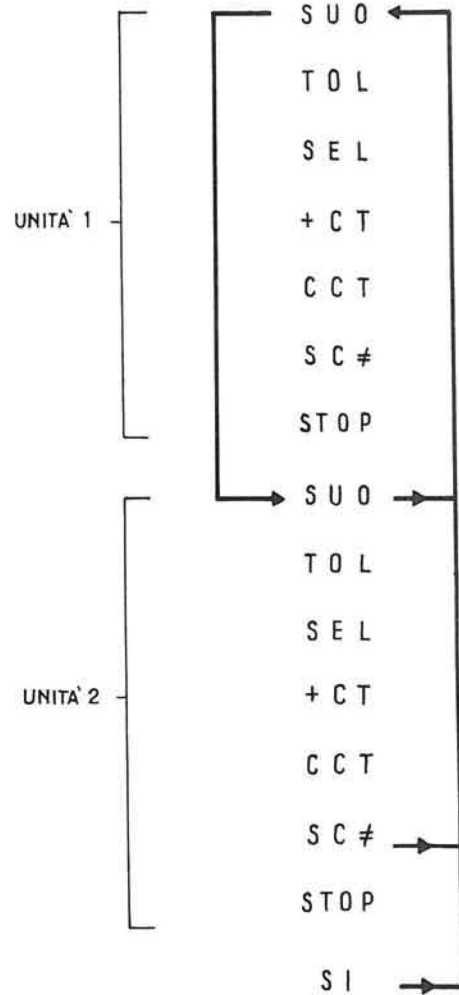
Le informazioni da perforare si trovano a partire dall'indirizzo 7000 e successivi in senso crescente.

Le informazioni lette sulle schede vengono registrate a partire dall'indirizzo 7500 e successivi in senso crescente.

Il programma occupa in memoria la zona compresa tra gli indirizzi 993 e 1128 (in totale 136 caratteri).

Il programma

La sequenza e' fornita da due parti: una aciclica e una ciclica.

SCHEMA DI 3° PROGRAMMA CON USO DELL'ISTRUZIONE S3P

ISTRUZIONE L	I	T	F	COD. SIMB.	INDIRIZZO ISTRUZIONE	RIFERIM.
, #	1 0 0 0	#	0	S 3 P	9 0 0 0	
# 0	7 0 0 0	1	9	C T	1 0 0 0	
# 0	7 5 0 0	2	9	C T	1 0 0 8	
% 1	1 0 7 2	#	0	S U O	1 0 1 6	
8 0	0 0 0 0	1	J	T O L	1 0 2 4	
al sottoprogramma di errore				S E L	1 0 3 2	
# 0	0 0 8 0	1	+	+ C T	1 0 4 0	
# #	7 5 6 0	1	5	C C T	1 0 4 8	
Y #	1 0 7 2	#	0	S C ≠	1 0 5 6	
σ 1	X X X X	#	0	S T O P	1 0 6 4	
% 2	1 0 1 6	#	0	S U O	1 0 7 2	
4 0	0 0 0 0	2	J	T O L	1 0 8 0	
al sottoprogramma di errore				S E L	1 0 8 8	
# 0	0 0 4 0	2	+	+ C T	1 0 9 6	
# #	8 3 4 0	2	5	C C T	1 1 0 4	
Y #	1 0 7 2	#	0	S C ≠	1 1 1 2	
σ 2	X X X X	#	0	S T O P	1 1 2 0	
) #	1 0 7 2	#	0	S I	1 1 2 8	

a) La prima parte e' composta da 2 istruzioni CT mediante le quali si portano gli indirizzi delle zone impegnate per la prima elaborazione esterna, rispettivamente nei registri T_1, T_2, (ad ogni ciclo si somma in T una quantita' pari al numero di caratteri per scheda).

b) La seconda comprende le due sequenze relative alle unita' in linea interessate.
Percorrendo per la prima volta le 2 sequenze si attivano le 2 unita' chiamate e si avvia, per ogni unita', un primo ciclo meccanico operativo.

Piu' precisamente :

Istruz. SUO — Non effettua il salto non essendo la unita' 1 occupata.
La SUO ha l'effetto di avviare la unita' e di predisporre il sincronizzatore al trasferimento che viene effettuato nella successiva istruzione TOL, da memoria dell'unita' centrale a memoria di transito dell'unita' 1.

Istruz. TOL — Trasferisce le informazioni relative alla prima scheda da perforarsi, da Memoria principale (a partire dall'indirizzo 0000 modificato da T_1 e successivi in senso crescente) a memoria di transito dell'unita' 1. Essendo 7000 il contenuto di T_1 l'indirizzo iniziale e' 7000.
Effettuato il trasferimento, si avvia un ciclo meccanico operativo dell'unita' 1, mentre l'unita' centrale passa ad eseguire contemporaneamente le successive istruzioni del programma.

Istruz. SEL

Se l'eventualita' si verifica salta al sottoprogramma di errore che provvede a ripristinare le condizioni esistenti prima del ciclo meccanico durante il quale l'errore si e' verificato, affinche' l'operazione possa ripetersi in modo corretto.

Istruz. +CT

Addiziona 80 (n^o caratteri di una scheda) al registro T_1.

Istruz. CCT

Confronta il contenuto T_1 con 560 (n^o totale dei caratteri da perforare ÷ 80). Dal confronto risulta che T_1 non e' uguale alla costante specificata nell'istruzione.

Istruz. SC≠

Effettua il salto all'istruzione SUO successiva, scavalcando lo STOP relativo all'unita' 1 perche' ancora mancano le condizioni d'arresto.

Il ciclo relativo all'altra unita' e' analogo a quello sopra descritto. Dal SC≠ dell'unita' 2 si ritorna alla SUO della unita' 1, e se e' ancora in corso il ciclo operativo di questa unita' la condizione di salto e' verificata; ma, essendo occupate in questo caso entrambe le unita' attivate, il salto non viene effettuato, e lo svolgimento del 3^o programma si arresta.

Arrestandosi il 3^o programma il primo prosegue a partire dall'istruzio-
che che immediatamente segue la S3P.

Il 1^o programma procede allora fino a che almeno una delle unita' occupate abbia terminato il ciclo opera-

tivo utile e si sia resa disponibile. Allorquando questa eventualita' si verifica il 1^{o} programma si arresta e il 3^{o} prosegue dal punto in cui si era arrestato.

Supponendo che l'unita' resasi disponibile sia la 2 si avra':

Istruz. SUO (relativa alla unita' 1)	L'unita' 1 e' ancora occupata, salta alla istruzione SUO relativa all'unita' 2.
Istruz. SUO (relativa alla unita' 2)	L'unita' 2 e' disponibile, Essendo l'unita' gia' attivata si predispone il sincronizzatore al trasferimento che viene effettuato nella successiva TOL.
Istruz. TOL	Trasferisce il contenuto della seconda scheda alla memoria di transito e quindi alla memoria principale. L'indirizzo iniziale e' ora 7500+40=7540.
Istruz. SEL	(solite condizioni).
Istruz. +CT	Addiziona 40 in T_3.
Istruz. CCT	Il confronto non da' ancora uguaglianza.
Istruz. SC≠	Essendosi avuta disuguaglianza dal precedente confronto, il salto viene effettuato.
Istruz. SUO (relativa alla unita' 1)	L'unita' 1 e' ancora occupata, si arresta il 3^{o} programma. Se l'unita' 1 si fosse nel frattempo resa disponibile, si sarebbe invece

percorso il rispettivo ciclo e l'arresto del 3° programma si sarebbe effettuato solo sulla SUO dell' unita' 2 .

Allorche' il contenuto di T_1, aumentato di 80 ad ogni + CT relativa, e' = 560 (quando cioe' tutte le 6 schede sono state perforate e ci si e' predisposti per una successiva scheda, percorrendosi il ciclo relativo all'unita' 1 si avra' :

Istruz. SC≠ — Verificatasi ora l'uguaglianza, l'istruzione STOP relativa all'unita' 1 non viene scavalcata.

Istruz. STOP — Arresta l'unita' 1 al termine dell'ultimo ciclo meccanico di lavoro .

Il programma prosegue passando al ciclo relativo alla unita' 2 sino a che anche le operazioni di lettura siano terminate.
L'esecuzione dello STOP dell'unita' 2 arrestera' infine il 3° programma.

10.8. Esempio di 3° sequenza con uso della S3P*

Generalita'

- Nell'esempio seguente si fa riferimento ad una sequenza di 3° programma che pur essendo avviata da una istruzione S3P, ha le singole unita' occupate mediante l'istruzione S3P*.
- La sequenza e' costituita da 2 cicli fondamentali riferentisi rispettivamente a 2 unita' : unita' 1 e 2 .
  L'unita' 1 e' supposta essere un perforatore di schede, l'unita' 2 una stampante in linea.

SCHEMA DI 3° PROGRAMMA CON USO DELL'ISTRUZIONE S3P *

- Le zone di memoria interessate dalle istruzioni TOL vengono preparate mediante sequenze d'istruzioni di 1° programma indicate sommariamente con la dicitura "Elabora zona SK (scheda)", " Elabora zona ST (stampa)".

- Ognuna di queste sequenze e' preceduta da un deviatore che viene aperto dal 3° programma ad effettuato trasferimento alla memoria unita' in linea di ogni scheda elaborata.

- I registri utilizzati T_1, T_2, T_3, T_4, associati a coppie alle unita' 1 e 2, servono per il posizionamento dei deviatori "α" e "β" del primo programma, "γ" e "δ" del terzo.

- Il numero delle schede da perforare e da stamparsi e' imprecisato ma determinabile mediante opportune istruzioni di programma raccolte sotto la descrizione generica "sono terminate le operazioni relative al 3° programma?".

Il programma

La sequenza di 3° programma e' composta da quattro parti di cui la prima e l'ultima acicliche.

a) La prima parte a cui rinvia l'istruzione S3P comprende due istruzioni TM che aprono rispettivamente i deviatori "γ" e "δ" per permettere la attivazione delle unita' in linea interessate nel corso dell'elaborazione.

b) La seconda parte e' introdotta dal deviatore "γ" che troviamo aperto ogni qualvolta dal 1° programma si esige la perforazione di una scheda.

c) La terza parte e' introdotta dal deviatore "δ" che troviamo aperto ogni qualvolta dal 1° programma si esige la stampa di una riga.

d) La quarta parte comprende due istruzioni di STOP relative alle due unita' in linea, condizionate all'alternativa di fine elaborazione posta nel 1° programma.

Si hanno pertanto le seguenti operazioni :

Istruzione S3P (1° programma) = rinvia all'istruzione T_5M e T_6M permettendo l'attivazione delle unita' in linea.

Istruzione T_5M (3° programma) = apre il deviatore "γ" permettendo il ciclo relativo alla unita' 1.

Istruzione T_6M = apre il deviatore "δ" permettendo il ciclo relativo alla unita' 2.

Istruzione SUO (≠) = (riferita ad unita' inesistente). Questa istruzione come si e' detto ha solo una funzione tecnica : su di essa termina ogni ciclo di 3° programma; da essa il ciclo riprende a segnalazione di unita' disponibile.

Istruzione SI o SN = funge da deviatore; viene alternativamente trasformata mediante apposite istruzioni in istruzione SI o SN.

Istruzione SUO = avvia l'unita' 1 e predispone il sincronizzatore al trasferimento.

Istruzione TOL = esegue un primo trasferimento da memoria principale a memoria di transito e avvia un ciclo meccanico di lavoro.

Il primo programma, come si e' detto, comprende 2 sequenze interessate rispettivamente alla preparazione dei dati da trasferire su schede perforate e a stampa.

In ognuna di esse, oltre che le istruzioni di elaborazione vera e propria, appaiono un deviatore, un posizionatore di deviatore, e una istruzione S3P*, con le seguenti funzioni :

deviatore "α" (1^a sequenza)	:	condiziona l'elaborazione di una nuova scheda all'avvenuto trasferimento della scheda precedente, nella memoria dell'unita' in linea. Puo' essere costituito da una istruzione SI che rinvia ad una istruzione qualsiasi di servizio (es. CT) che la preceda immediatamente.
posizionatore "γ" (1^a sequenza)	:	apre il deviatore "γ" che permette l'entrata in ciclo della sequenza di 3^o programma relativa all'unita' 1.
istruzione S3P* (1^a sequenza)	:	avvia un ciclo di 3^o programma con conseguente esecuzione della sequenza relativa all'unita' 1.

Il deviatore β , il posizionatore δ e la S3P* della seconda sequenza hanno funzioni analoghe a quelle su descritte, ma relativamente all'unita' 2.

Opportune istruzioni di 1^o programma permettono infine l'avvio del 3^o programma sui 2 STOP che ne condizionano l'arresto.

Lo STOP di 1^o eseguito immediatamente dopo la S3P* arresta invece il primo programma.

Istruzione SEL = in caso di errore, salta al sottoprogramma di errore che provvede a riprodurre le condizioni esistenti prima del ciclo meccanico durante il quale l'errore si e' verificato, affinche' l'operazione possa ripetersi in modo corretto, (le caratteristiche dell'istruzione SEL sono esposte nello schema relativo).

Istruzione T_1M = trasforma l'istruzione "deviatore" (SI o SN) precedentemente considerata, in salto effettivo; chiude cioe' il deviatore "γ" che puo' essere riaperto solo da una istruzione posta nel 1^o programma.

Istruzione T_2M = stessa funzione della T_1M, ma relativamente al deviatore "α" posto nel 1^o programma il cui posizionamento condiziona l'elaborazione di una eventuale scheda da perforare.

Istruzioni : SI, SUO, TOL, SEL, T_3M, T_4M (relative alla unita' 2) = per queste istruzioni valgono le considerazioni fatte per il gruppo precedentemente descritto ma relativamente all'unita' in linea N^o 2: stampante in linea.

Istruzione STOP (1^o unita')
Istruzione STOP (2^o unita') } lette di seguito mediante S3P* di 1^o programma permettono lo arresto del 3^o programma.

CAP. 11° : IL TAVOLO DI COMANDO

11.1. Generalita'

Il tavolo di comando e' l'organo che costituisce un tramite di comunicazione fra l'operatore ed il sistema per l'elaborazione dei dati. Con esso si puo' seguire lo svolgersi di tutte le operazioni, ed eventualmente intervenire in tale svolgimento.

Il quadro di comando e di controllo contiene i tasti di comando con i quali si puo' agire sulla calcolatrice dall'esterno, e batterie di indicatori che indicano lo stato di avanzamento delle elaborazioni oppure localizzano ed individuano eventuali errori.

La funzione specifica dei tasti di comandi e degli indicatori luminosi sara' piu' ampliamente trattata nelle pagine seguenti.

Al tavolo di comando e' connessa una Telescrivente, che ha l'ufficio di prelevare il contenuto delle posizioni di Memoria che si desidera indagare dandone la trascrizione a stampa su di un foglio di carta.

Si ha cosi' la possibilita' di ottenere per iscritto qualche risultato intermedio particolarmente interessante o significativo, senza attendere la fine della elaborazione e la stampa finale nella stampante.

La telescrivente e' in grado di stampare caratteri alfanumerici e speciali alla velocita' di 360 caratteri/min. con 72 caratteri al max per riga. La telescrivente torna automaticamente a capo all'inizio di ogni istruzione che segue una stampa oppure quando sia necessario stampare piu' di 72 caratteri mediante una sola istruzione.

Un perforatore di banda collegato alla telescrivente consente di ottenere contemporaneamente alla scrittura la

perforazione di banda di carta. La telescrivente e' fornita anche di una tastiera, che si utilizza ogni qualvolta si desideri portare dei caratteri in Memoria, impostandoli direttamente sulla tastiera stessa.

Al tavolo di comando e' pure connesso un lettore di nastro perforato per mezzo del quale e' possibile introdurre dati direttamente in Memoria. Il lettore e' particolarmente utile per l'introduzione di programmi da mettere a punto, eliminando la fase di conversione su nastro magnetico.

Terminata la fase di correzione il programma in via normale viene definitivamente registrato su nastro magnetico, da dove sara' successivamente prelevato per la esecuzione vera e propria del lavoro.

11. 2. Quadro di comando manuale

Il quadro di comando manuale e' composto di una serie di tasti che per comodita' raggruppiamo in 6 zone.

5
TAMB. TS. FOT. LETT. V.FOT. STOP FOT.
6
A.S.E. A.E. A.S.N. A 'S. SEP. ISOL.
4
1 2 3 E2 E3 E4
3
MAN. AUTO. SING. CO P.
1
P8 P7 P6 P5 P4 P3 P2 P1 BL
0 1 2 3 4 5 6 8 9 ~ + ≠ / - ÷
φ A B C D E F H I δ = α % , γ
E J K L M N O P Q R * (β $ η π
· S T U V W X Y Z ' ⊖ !) ? & ⊗
2
SGL
VIA

Zona 1 : Pulsanti P1 ÷ P8, tastiera e pulsante BL.

P8	P7		P6	P5	P4	P3		P2	P1			BL			
0	1	2	3	4	5	6	7	8	9	~	+	≠	/	-	÷
φ	A	B	C	D	E	F	G	H	I	δ	=	α	%	,	γ
ε	J	K	L	M	N	O	P	Q	R	*	(	β	$	η	π
.	S	T	U	V	W	X	Y	Z	'	θ	!	)	?	&	⊗

L'elaboratore oltre che eseguire le istruzioni di programma registrate in memoria, esegue pure istruzioni impostate direttamente sul quadro di comando.

Per questo esiste una tastiera di 64 tasti corrispondenti ai 64 caratteri usati dalla calcolatrice e otto tasti cosi' segnati :

P1, P2, P3, P4, P5, P6, P7, P8.

Per impostare un'istruzione sul quadro di comando e' sufficiente premere il tasto P8 e quindi premere sulla tastiera gli otto caratteri relativi all'istruzione stessa.

Il primo carattere impostato viene registrato in ottava posizione (P8), il secondo in P7 e cosi' via.

Qualsiasi battuta in tastiera dopo l'ottava (P1) non ha alcun effetto.

Se si vuole correggere un'istruzione gia' impostata e' sufficiente premere il tasto P corrispondente alla posizione da modificare e battere in tastiera i caratteri sostitutivi.

Il pulsante BL esclude la tastiera. Con l'abbassamento di questo tasto si annulla l'effetto di ogni battuta su tastiera.

Zona 2 : Tasti VIA, SG, SGL.

Serve ad annullare tutte le condizioni automatiche relative alla terza sequenza di programma che compaiono sul quadro di controllo nella zona 4.

Serve ad annullare tutte le condizioni automatiche di macchina che appaiono sul quadro di controllo nelle zone 10, 11, 12, 13 .

Premendo il tasto VIA il calcolatore inizia le operazioni con le modalita' dipendenti dal posizionamento dei tasti della zona 3.

Zona 3 : Tasti MAN, AUTO, SING, CONT, PP.

MAN. AUTO. SING. CONT. P.P.

Col tasto PP abbassato, premendo il tasto VIA, si esegue una istruzione procedendo di cifra per periodo di cifra.

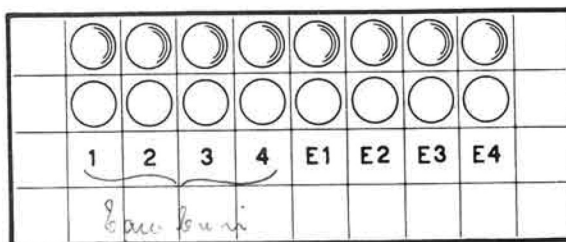
Col tasto CONT abbassato, premendo il tasto VIA si esegue la prima istruzione e quindi automaticamente tutte le altre fino alla fine del programma.

Col tasto SING abbassato, premendo il pulsante VIA si esegue una istruzione alla volta e la calcolatrice si ferma al termine di ogni istruzione.

Col tasto AUTO abbassato, premendo il tasto VIA si eseguono le istruzioni del programma passo passo, una per volta, oppure tutto il programma a seconda che siano rispettivamente abbassati i tasti PP, SING, CONT.

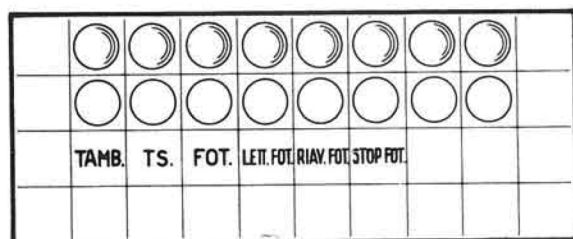
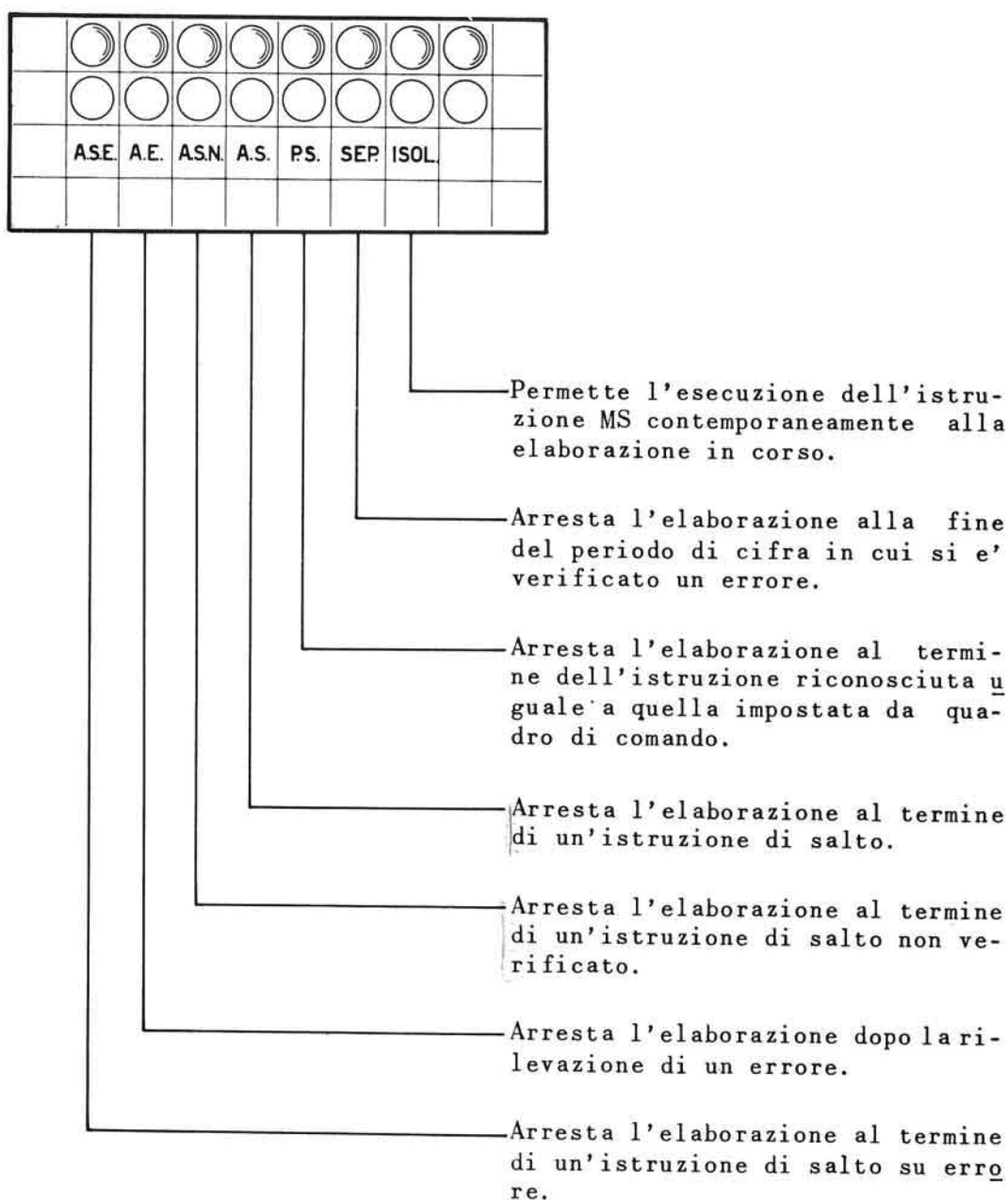
Col tasto MAN abbassato, premendo il pulsante VIA si esegue l'istruzione impostata su quadro di comando che viene eseguita periodo di cifra per periodo di cifra o interamente a seconda che siano abbassati i tasti PP o SING.

Zona 4 : Tasti 1, 2, 3, 4, E1, E2, E3, E4.

1	2	3	4	E1	E2	E3	E4

I selettori "Condizioni esterne" E1, E2, E3, E4, se abbassati abili tano il salto rispettivamente per le istruzioni, sia da console che di programma registrato in memoria, SE1, SE2, SE3, SE4.

Per passare dalla posizione "Salta" a quella "Non salta" e' sufficiente premere a fondo i singoli tasti.

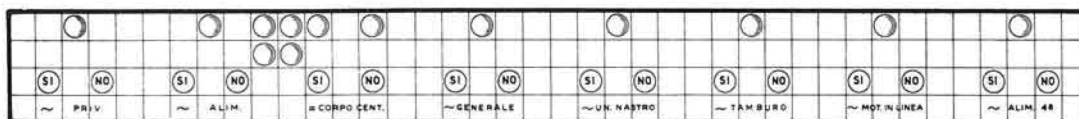
I selettori 1, 2, 3, 4 se abbassati permettono il verificarsi delle condizioni esterne per il tamburo.

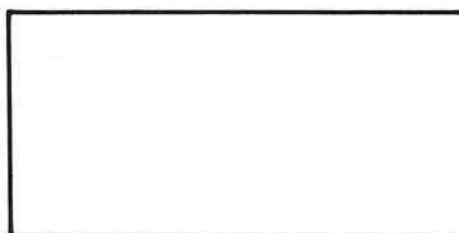
Zona 5 : Tasti TAMB, TS, FOT, Lett.Fot, Riav.Fot, StopFot

Zona 6 : Tasti ASE, AE, ASN, AS, PS, SEP, ISOL.

I tasti non considerati hanno una funzione variabile a seconda delle particolari caratteristiche dell'impianto.

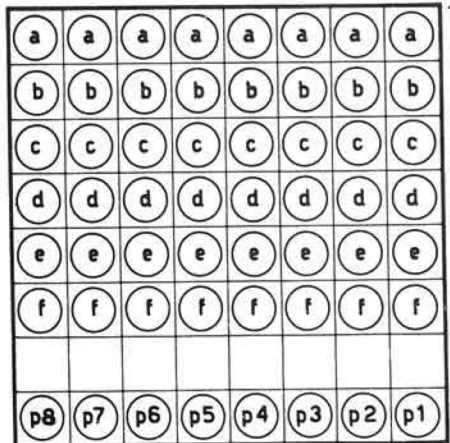
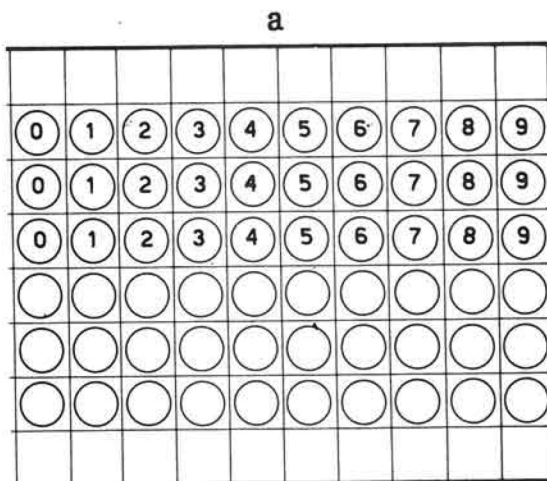
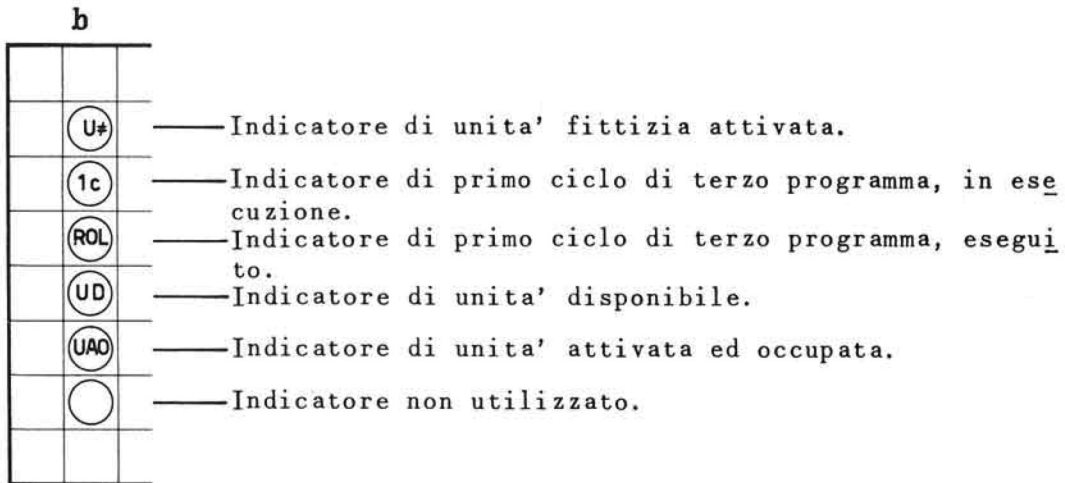
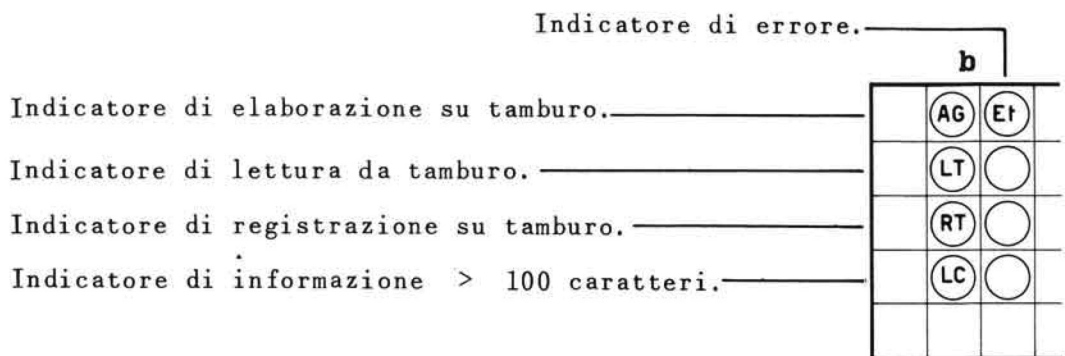
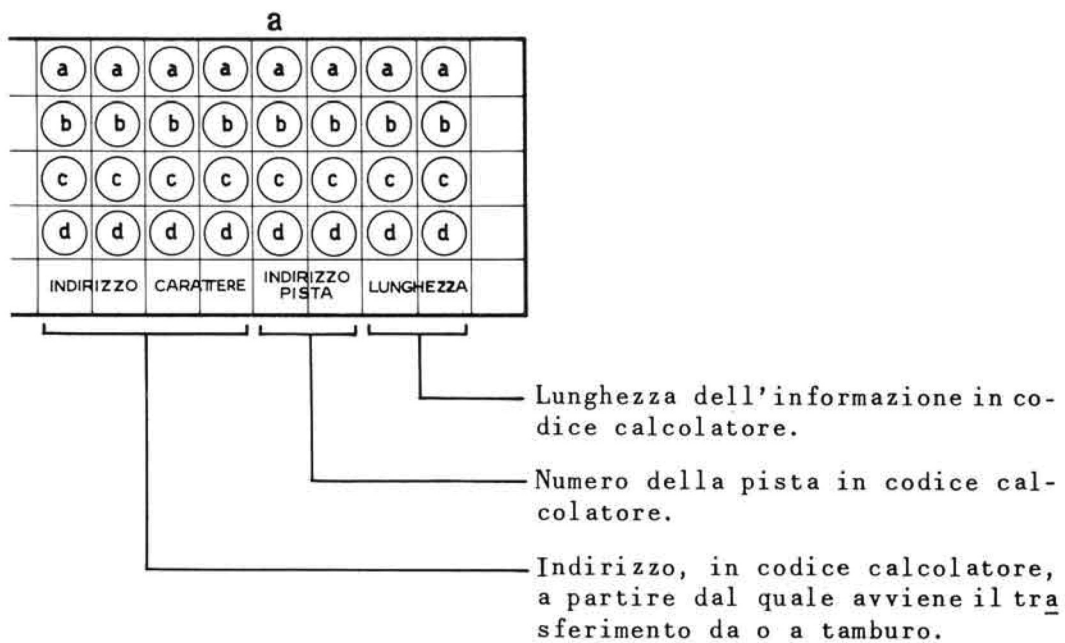
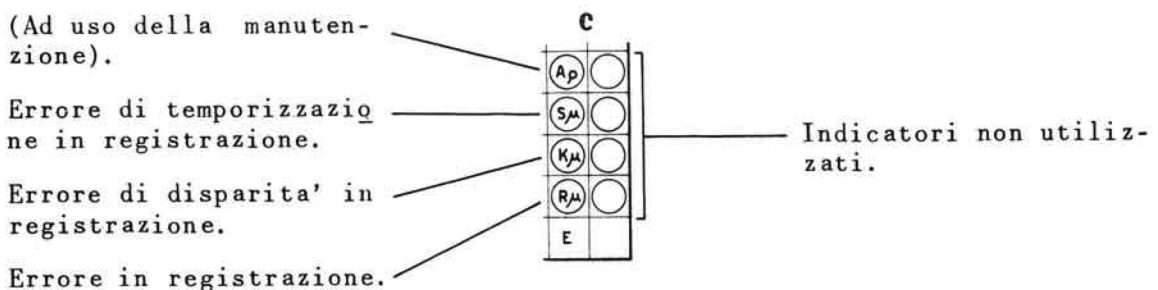
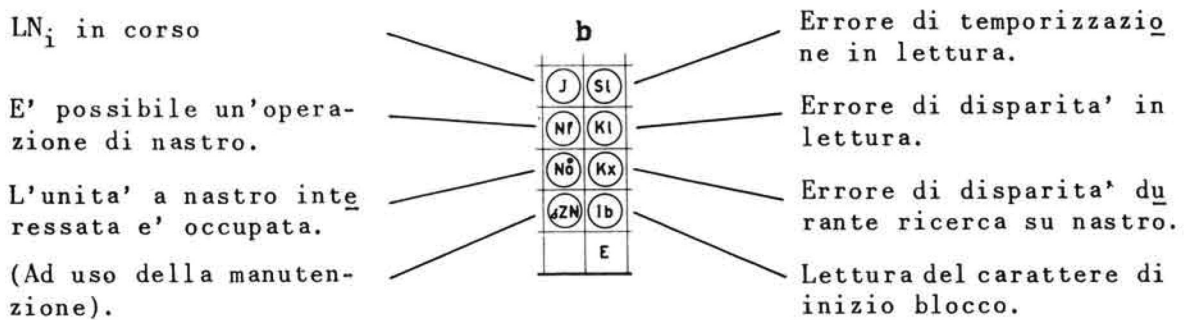
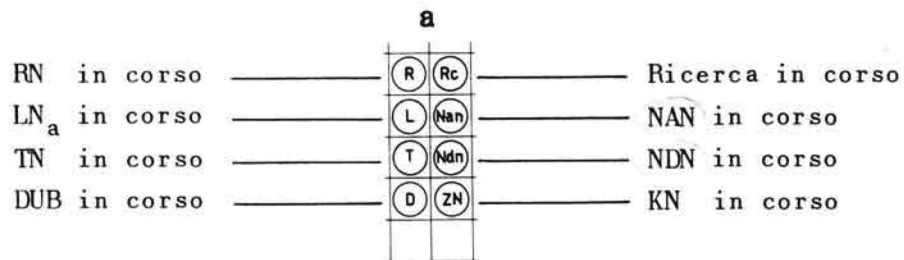
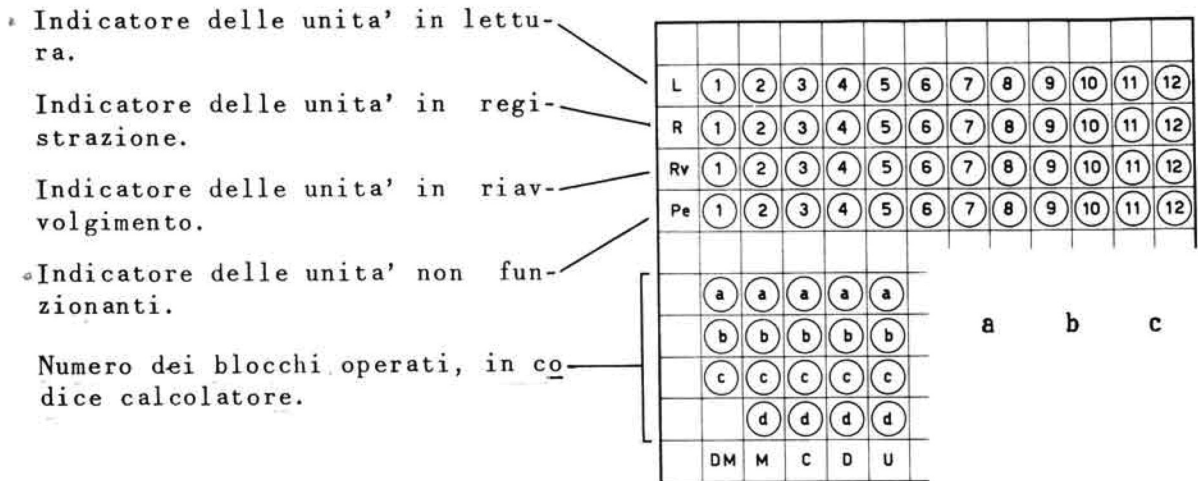
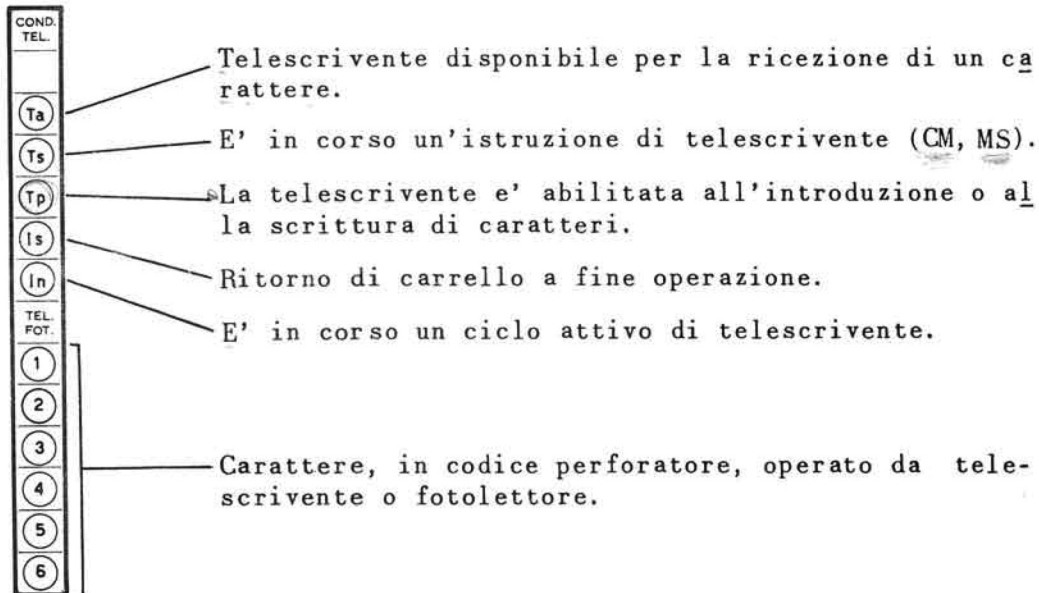
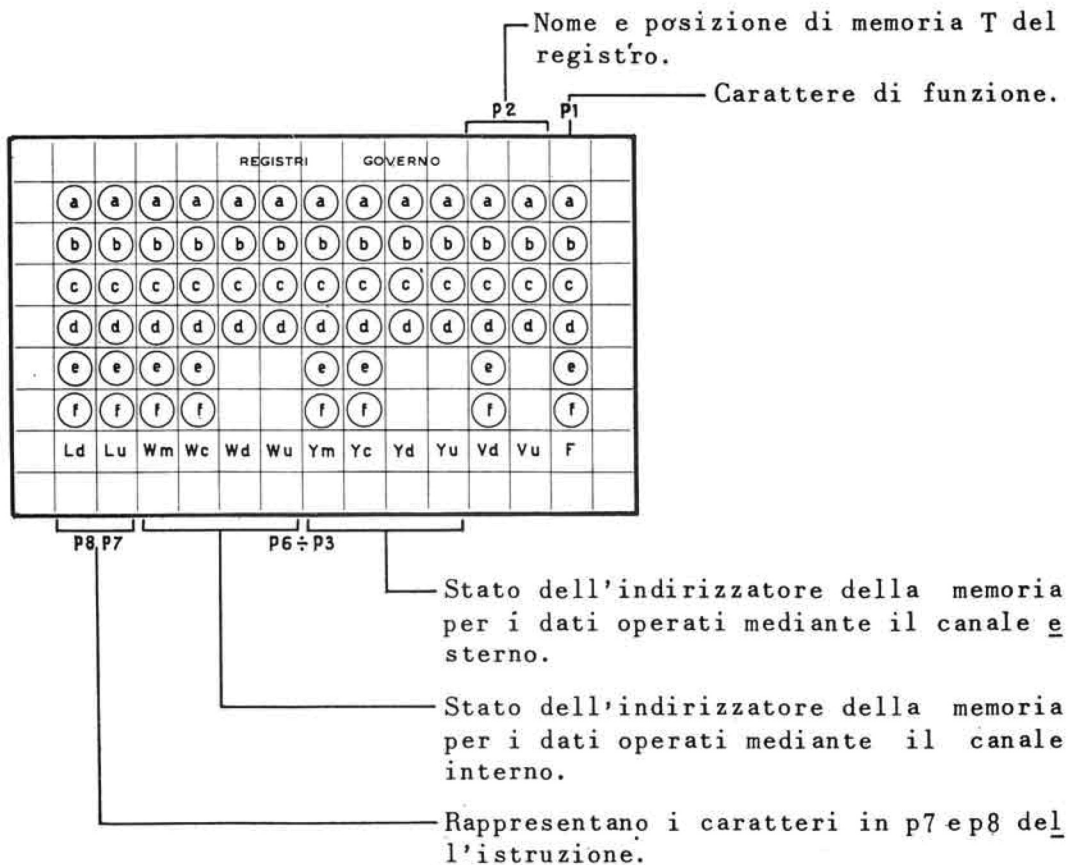
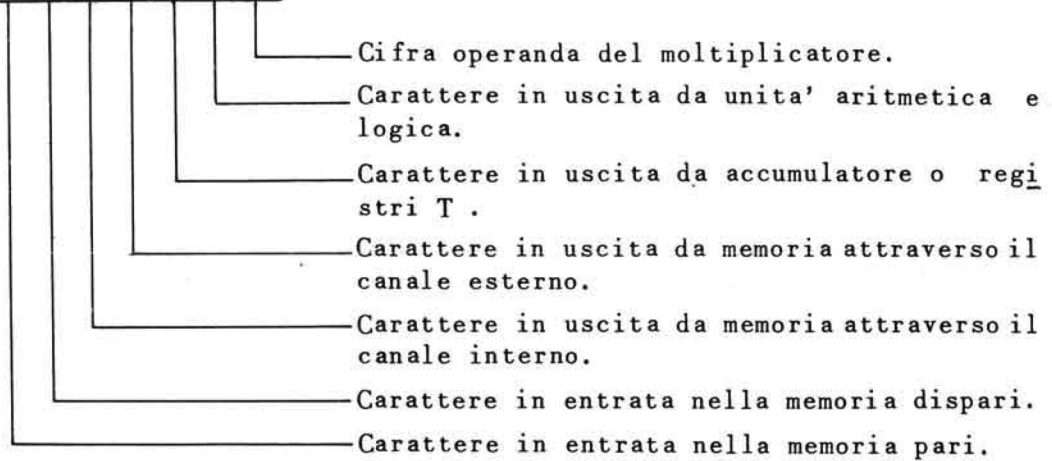
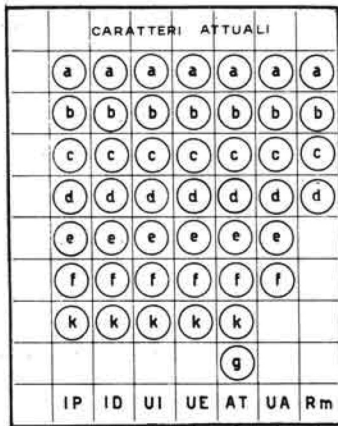
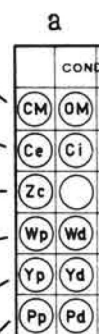
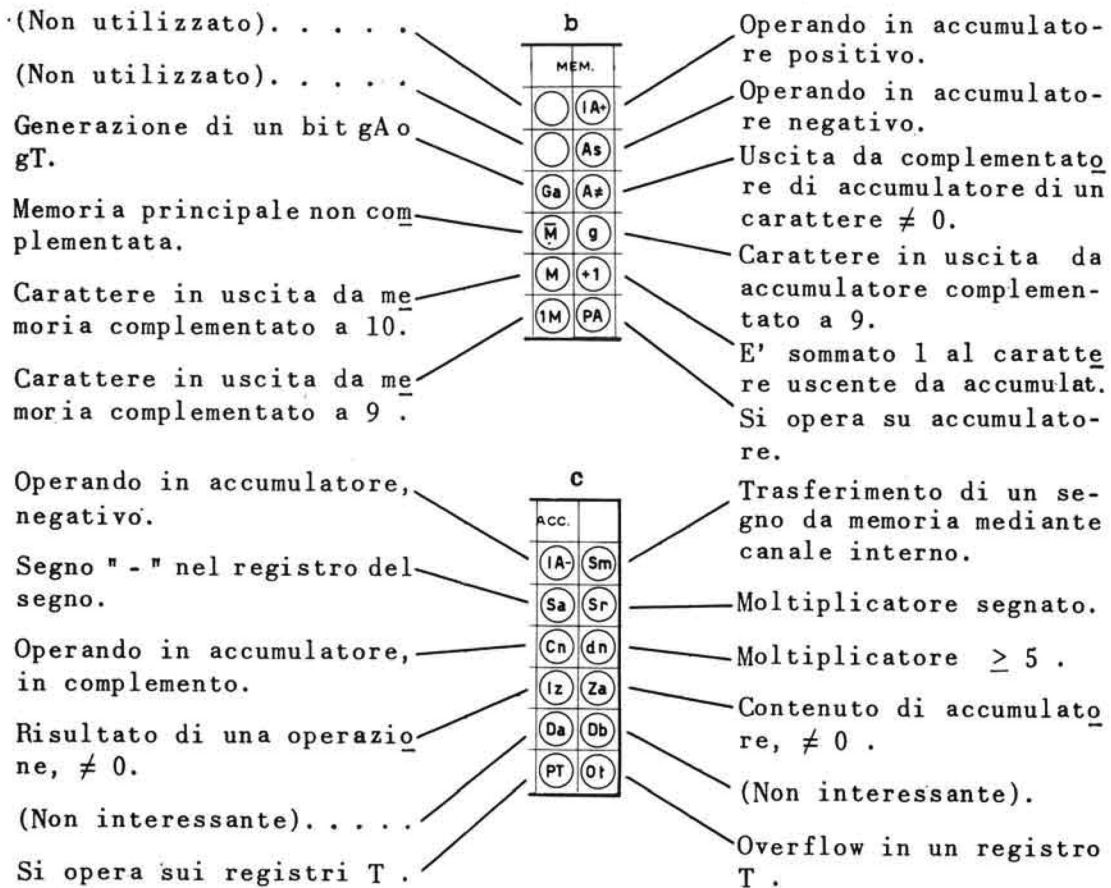
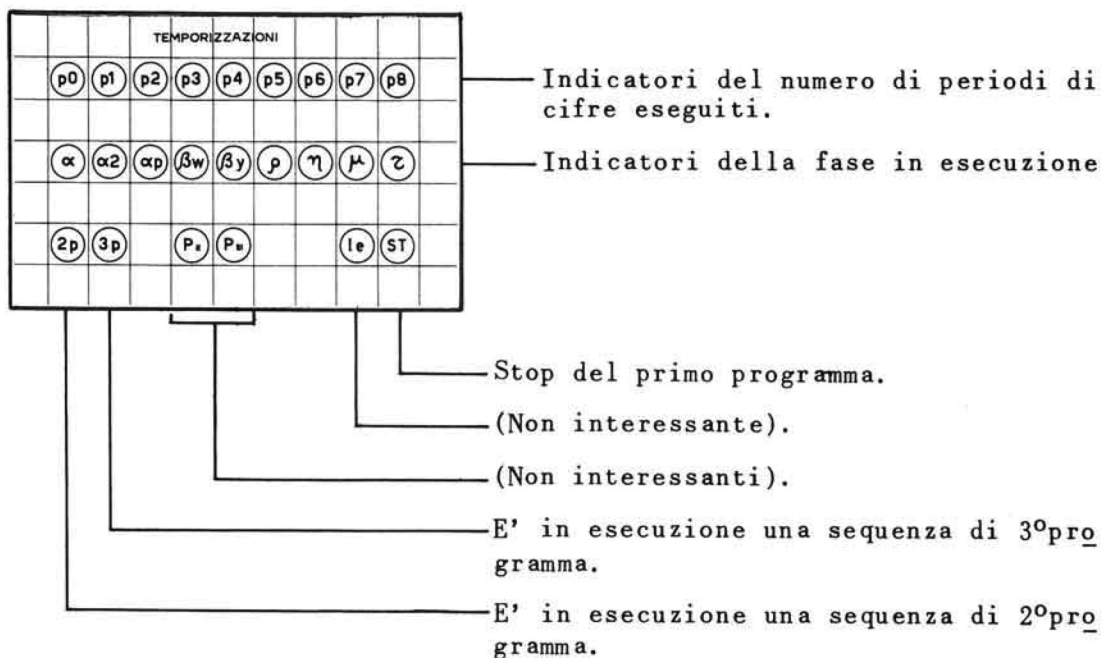
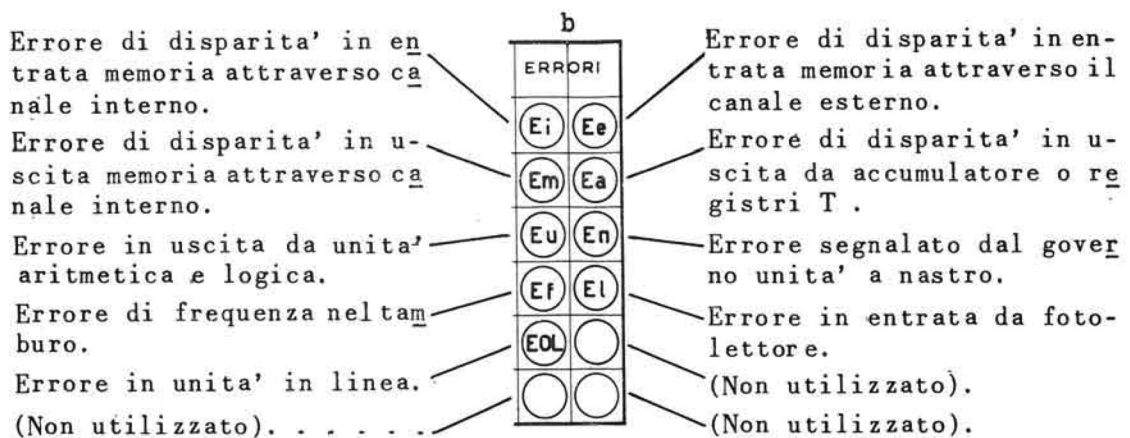
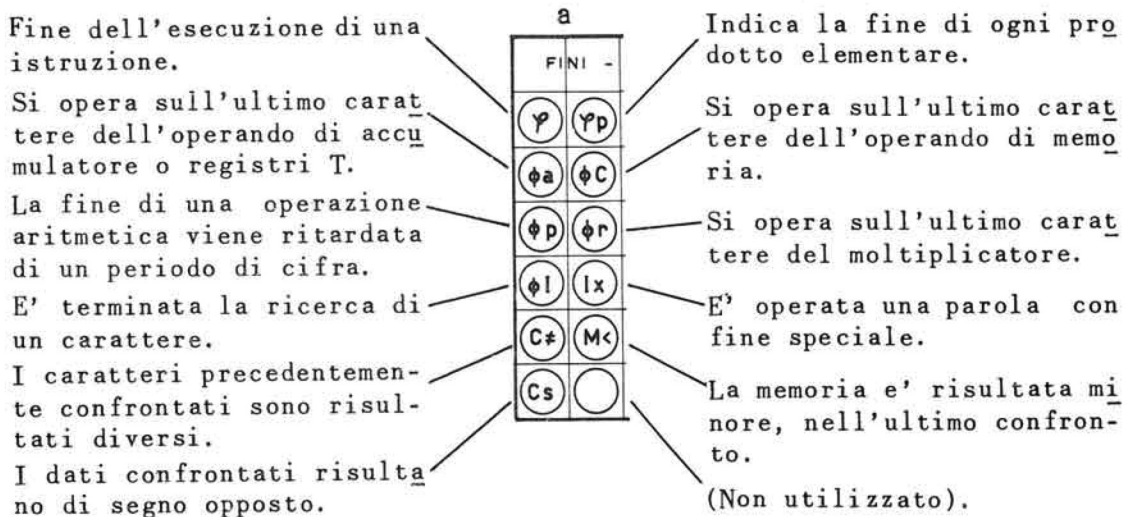
11.3. Il quadro di controllo

Il quadro di controllo si compone di 13 batterie di indicatori luminosi poste superiormente al quadro di comando manuale.

Ad ogni batteria corrisponde una delle zone sottoindicate.

Zona 1 : Riquadro di alimentazione

~ PRIV.	Interruttore corrente alternata privilegiata del corpo centrale.
~ ALIM.	Interruttore corrente alternata alimentatori.
= CORPO CENTR.	Interruttore corrente continua del corpo centrale.
~ GENERALE	Interruttore corrente centro elettronico.
~ UN. NASTRO	Interruttore corrente del governo unita' nastro.
~ TAMBURO	Interruttore corrente del tamburo.
~ MOT. IN LINEA	Interruttore corrente alternata delle unita' in linea
~ ALIM. 48	Interruttore corrente delle unita' in linea.

Zona 2 :

In questa zona possono apparire dispositivi diversi secondo le necessita' particolari dell'utente.

Zona 3 : Indicatori "Istruzione manuale" ; permettono il controllo delle istruzioni impostate manualmente

Zona 4 : Indicatori relativi alla terza sequenza di programma.

Zona 5 : Indicatori relativi al tamburo magnetico.

Zona 6 : Indicatori relativi alle unita' a nastro magnetico.

Zona 7 : Indicatori relativi alla telescrivente e fotolettore.

Zona 8 : Indicatori della configurazione in codice calcolatore dell'istruzione in corso.

Zona 9 : Indicatori dei caratteri in corso di elaborazione

Zona 10 : Indicatori delle condizioni della memoria, dell'accumulatore e dei registri T.

Zona 11 : Indicatori di temporizzazioni

Zona 12 : Indicatori di fini ed errori

Zona 13 : Indicatori relativi a condizioni varie

COND.	VARIE
(Non utilizzato).	Ar: (Non interessante).
Si: L'istruzione in esecuzione e' diversa da quella impostata da quadro di comando	No: L'unita' a nastro interessata e' occupata.
CV: Si e' verificata la condizione di un'istruzione di salto.	Kz: Indicatore di passaggio tra un periodo di cifra e l'altro.
Ni: E' bloccato il primo programma.	Ne: E' bloccato il secondo programma.
N3: E' bloccato il terzo programma.	λ: (Non interessante).
Sb: E' terminata l'istruzione bloccante del secondo programma.	Sb3: E' terminata l' istruzione bloccante del terzo programma.